

Universidad Complutense de Madrid

Facultad de Informática

Máster en Investigación en Informática

Proyecto Fin de Master en Sistemas Inteligentes

(Curso 2008-2009)

Controlador Fuzzy de un Quadrotor

AUTOR: Francisco Morata Palacios

DIRECTOR: Matilde Santos Peñas

Resumen

El presente trabajo desarrolla la implemetación de un modelo que utiliza un controlador fuzzy para estabilizar el funcionamiento de un cuadrotor. Un cuadrotor es un helicóptero con cuatro rotores, el cual permite tener una mayor estabilidad que un helicóptero normal. El cuadrotor como cualquier otro cuerpo rígido posee seis grados de libertad, tres que definen su posición: altura, desplazamiento vertical y desplazamiento horizontal; y otros tres que definen la orientación del cuerpo: alabeo, cabeceo y guiñada. Para la estabilización se utiliza un controlador fuzzy al que como entradas le llegan la altura deseada y los tres giros que se quieren hacer y como salida se obtendrá la potencia necesaria en los motores para alcanzar esa situación desde su estado actual. La simulación del modelo se ha hecho con la herramienta Simulink de Matlab con tiempos de respuesta y prestaciones muy buenas.

Palabras Claves

Cuadrotor, estabilización de helicópteros, lógica difusa, controlador fuzzy, controlador PID, simulación, Simulink.

Abstract

The aim of this text is to describe the implemetation of a cuadrotor model with fuzzy logic. Cuadrotor is an helicopter with four rotors, that makes the machine have more stability than in a normal one. Cuadrotor has six libertity grades, three of them due to the position: height, horizontal movement, vertical movement; and the other three about the orientation: pitch, roll and yaw. A fuzzy controlator is used to established the model. It has four entries: desired height, desired roll, desired pitch and desired yaw; the exits are the potency of each rotor necesary to reach the desired situation from the present state. Simulation has been made succesfully with Simulink, tool of Matlab, with excelent times.

Key Words

Cuadrotor, fuzzy logic, fuzzy controlator, PID controlator, simulation, Simulink.

Índice de contenido

1.Introducción. Motivación y Objetivos.....	2
2.Estado del Arte.....	3
2.1. Control por Realimentación. El Controlador PID.....	3
2.2. Lógica Fuzzy.....	6
2.2.1. Fundamentos de Lógica Fuzzy.....	8
2.2.2. Controlador Fuzzy.....	10
2.3. Planta del quadrotor.	11
2.3.1.Descripción.....	11
2.3.2. Cinemática.....	11
2.3.3. Dinámica.....	13
2.3.4. Movimientos Rotacionales.....	14
2.4. Descripción del Modelo	14
2.4.1. Dynamics.....	14
2.4.2. Rotor.....	15
2.4.3. Ini_Cond.....	15
2.4.4. Integrator.....	15
2.4.5. Physical Limitations.....	15
2.4.6. Sc.....	16
2.4.7. Control.....	16
2.4.8. Desired State.....	16
3.Controlador Fuzzy.....	16
3.1. Módulo Agregación.....	22
3.2. Control Fuzzy PID.....	25
3.2.1. Control Fuzzy PID Altura.....	26
3.2.2. Control Fuzzy Roll.....	27
3.2.3. Control Fuzzy Pitch.....	28
3.2.4. Control Fuzzy Yaw.....	29
4.Resultados.....	31
4.1. Resultados Individuales.....	31
4.1.1. Resultados de la Altura.....	32
4.1.2. Resultados del Roll.....	47
4.1.3. Resultados del Pitch.....	33
4.1.4. Resultados del Yaw.....	36
4.2. Resultados Combinados.....	37
4.2.1. Primera Prueba.....	39
4.2.2. Segunda Prueba.....	41
5.Conclusiones y trabajos futuros.....	44
Bibliografía	45

Introducción. Motivación y Objetivos.

Los helicópteros autónomos son una de las plataformas más implantadas en robótica debido a las funcionalidades de despegue, aterrizaje vertical y vuelo estacionario que se pueden conseguir. Estas funcionalidades son muy útiles en desastres naturales, búsquedas y rescates, vigilancia, inspección de instalaciones y estructuras, vigilancia y realización de mapas de terreno (Ollero and Maza 2007). En muchas de las ocasiones las misiones entrañan riesgos o requiere una gran movilidad o bien la aeronave es necesario que tenga una dimensión mínima, lo que ahonda en la idea de que no vaya tripulado, sino que sea automático.

Este trabajo va a tratar el caso de un **quadrotor**, es decir un helicóptero con cuatro rotores en lugar de uno. El quadrotor tiene como ventaja principal la estabilidad del aparato, como desventaja la dificultad que entraña coordinar los cuatro rotores a la vez.

El quadrotor que se presenta en este trabajo está formado por un sistema realimentado desarrollado con un sistema de control que indica en cada momento las acciones a tomar por cada uno de los motores que hacen funcionar a los rotores.

El controlador propuesto utiliza la técnica de lógica fuzzy (difusa). Infinidad de productos lanzados al mercado usan tecnología difusa, muchos de ellos utilizando la etiqueta fuzzy como símbolo de calidad y prestaciones avanzadas. El control difuso ha sido exitosamente aplicado en procesos industriales y tecnológicos en empresas destacadas como Sony, Cannon, Panasonic, Bosch, Mazda, y Nissan.

El presente documento está organizado de la siguiente manera: en la sección 2 se hace una introducción al estado del arte del trabajo. En la sección 3 se expone el controlador difuso implementado. En la sección 4 se muestran los resultados de todos los movimientos con pruebas individuales y con pruebas completas que combinan distintos movimientos. Finalmente en la sección 5 se presentan las conclusiones del trabajo expuesto.

Estado del Arte

En este apartado se van a explicar todos los elementos que se han tenido que estudiar y usar para poder desarrollar el controlador.

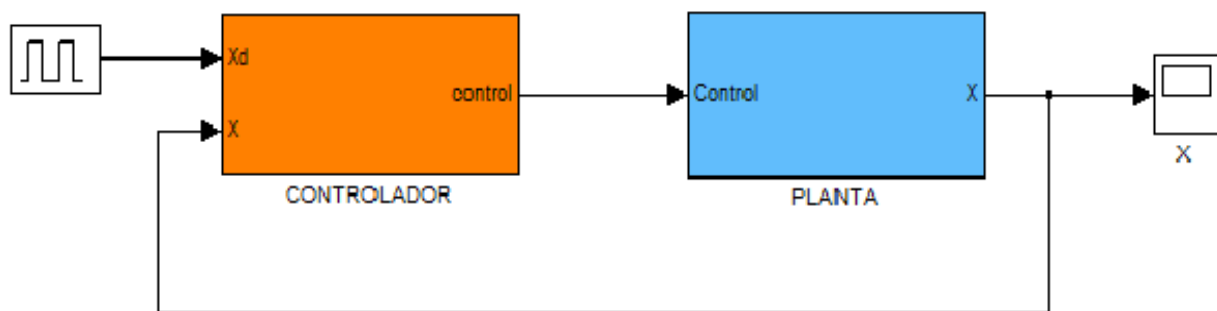
En el subapartado 1 se hace una introducción general a los sistemas realimentados. En el subapartado 2 se desarrolla la teoría fuzzy necesaria para comprender el controlador. En el subapartado 3 se describe las principales características del quadrotor y las ecuaciones físicas que simulan su comportamiento. En el subapartado 4 se describe el modelo dinámico implantado en el cual se debe integrar el controlador desarrollado.

2.1. Control por Realimentación. El Controlador PID

Una de las técnicas más utilizadas en el control de procesos, consiste en utilizar una transformación no lineal para obtener un sistema resultante lineal. Sobre las cuales se utilizan técnicas clásicas de control, como los PID.

Un sistema realimentado es aquel en el cual la salida del sistema se vuelve a introducir adicionándola a la entrada. El controlador PID clásico ha sido ampliamente utilizado y lo usaremos como punto de referencia. La mayoría de los controladores empleados en la industria son PID o variantes del mismo.

La idea del control realimentado es emplear precisamente esta característica para conseguir que la respuesta del sistema sea la que se desee. El simple hecho de realimentar un sistema, no es suficiente para conseguir esto. La razón es, que normalmente los parámetros de la función de transferencia de un sistema vienen dados por la dinámica del sistema y no se pueden alterar, por tanto, no es posible, simplemente realimentando el sistema, conseguir cualquier salida que deseemos. Una solución posible, – bastante extendida –, es añadir a al sistema original $G(t)$ un segundo sistema a la entrada $C(t)$, cuyos parámetros podamos ajustar a voluntad. Este segundo sistema recibe el nombre de **controlador** y al sistema original se le da el nombre de **planta**. De este modo, el controlador recibe como entradas, la diferencia entre un valor de referencia $r(t)$ que es el valor que se desea obtener a la salida de nuestra planta, y la salida real de la planta $y(t)$. La respuesta del controlador tendrá que ser tal, que al cabo del tiempo consiga hacer que $r(t) - y(t) = 0$. (Ñeco, 2003)



Modelo General de un Sistema Realimentado

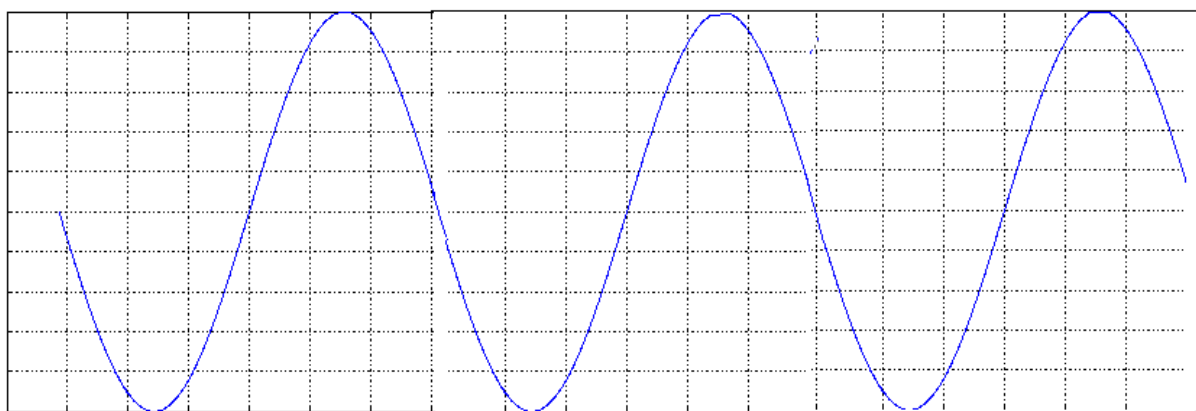
La entrada al controlador $e(t)$ recibe el nombre de señal de error ya que representa la diferencia entre el valor de referencia y la salida real de la planta. La salida del controlador, recibe el nombre

de señal de control $c(t)$, siendo esta señal la nueva entrada a la planta. La cuestión es como diseñar el controlador para conseguirla salida deseada o, lo que es lo mismo, para conseguir que la señal de error valga cero. La solución más sencilla posible es considerar un controlador cuya salida sea directamente **proporcional** al error (**controlador P**), es decir si la salida de la planta es menor que la señal de referencia la señal de control será positiva y proporcional a $r(t) - y(t)$ y si por el contrario fuera mayor, la señal de referencia será ahora negativa y proporcional a $r(t) - y(t)$. A este tipo de controlador se le llama proporcional. El control proporcional no es, en general, suficiente para controlar una planta por varios motivos. El primero de ellos, es que, cuando la planta ha alcanzado el valor de consigna, la señal de control se hace cero. Para muchos sistemas esto no es una entrada válida, lo que acabaría provocando una *oscilación* en la entrada del sistema. Pero además, para muchos sistemas, ni siquiera es capaz de hacer que la planta alcance el valor deseado.

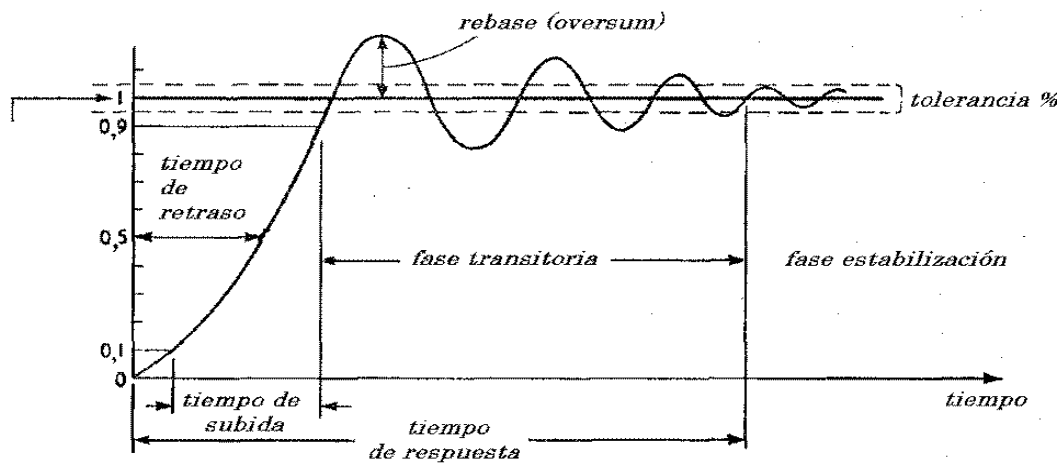
Una solución es añadir un segundo término proporcional a la **integral** del error, de alguna manera, este segundo término actúa como una "memoria" que lleva cuenta de la historia previa del error. Cuando se añade al controlador, evita que este alcance un valor de equilibrio para un error dado, como en el proporcional (**controlador PI**).

Si al controlador PI anteriormente descrito, se le añade un término que sea proporcional a la **derivada** del error, El resultado es un **controlador PID** completo. Un término proporcional a la derivada del error significa que, si este crece con el tiempo, su derivada será positiva y tanto mayor cuanto más rápido crezca. Si por el contrario decrece, la derivada será negativa y, de nuevo, tanto más grande en valor absoluto cuanto más rápido decrezca. Esta propiedad, hace que el término derivativo de alguna manera se anticipe al valor del error, puesto que su acción depende del ritmo al que este está variando.

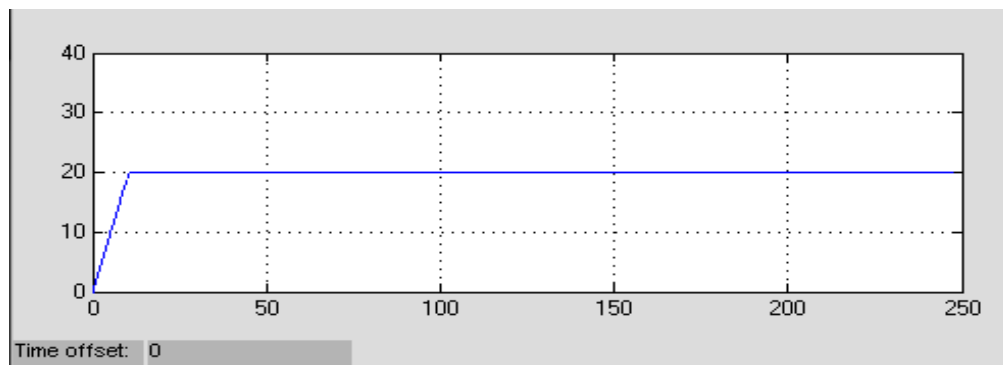
Como complemento a esta explicación se deben señalar los tipos de respuestas que pueden dar estos sistemas. El peor de todos es el comportamiento **inestable** que se produce cuando la respuesta no tiene un carácter definido y no se acerca al valor deseado. Luego están los **estables**, hay varios, está el que tiene una estabilidad crítica oscilante de amplitud constante en el que es el que el controlador no consigue llegar al valor deseado a lo largo del tiempo, pero oscila alrededor del valor (figura 2). Mejor es el estable con salida oscilatoria amortiguada, en el que el sistema si que consigue corregir el error en un determinado tiempo (figura 3). Por último está la respuesta amortiguada crítico estable en la que no se producen rebases ni oscilaciones (figura 4). Es la ideal, pero se debe tener cuidado que el tiempo de subida sea muy grande quedando sobreamortiguada. Una de las ventajas de los controladores fuzzy es que tienen más facilidad que los clásicos en obtener este tipo de respuesta.



Respuesta con estabilidad critica-oscilante



Respuesta con salida estable amortiguada



Respuesta amortiguada crítico-estable

2.2. Lógica Fuzzy

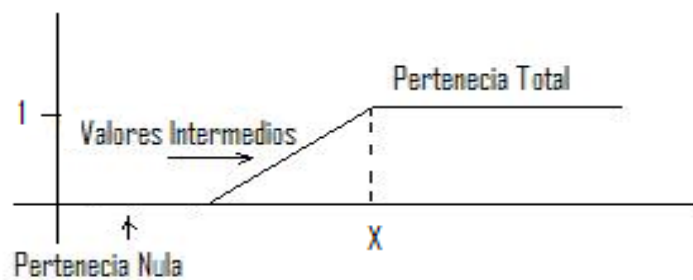
La lógica difusa o fuzzy se presenta como una técnica de la Inteligencia Artificial cuyo objetivo es expresar el conocimiento común, el cual es de tipo lingüístico cualitativo, en un lenguaje matemático regido por la teoría de conjuntos difusos y funciones de pertenencia asociados a estos. De esta manera relaciona la sintaxis lingüística, la cual es de mayor entendimiento para las personas, con datos numéricos que aportan mayor precisión.

La aplicación de la lógica difusa al modelo y control de problemas no lineales, presenta virtudes en su flexibilidad y tolerancia con las imprecisiones, todo esto basado en el lenguaje natural.

Una de las grandes ventajas de la lógica fuzzy es que a medida de que la complejidad de un sistema aumenta, disminuye nuestra capacidad para hacer afirmaciones precisas. En esto, dado su naturaleza, la logica fuzzy es un excelente método cuando los datos no son precisos.

2.2.1. Fundamentos de Lógica Fuzzy

La lógica fuzzy o borrosa está formada por conjuntos borrosos. Un conjunto borroso es un conjunto en donde la pertenencia de sus elementos se define mediante una función denominada *función de pertenencia*.



Conjunto difuso

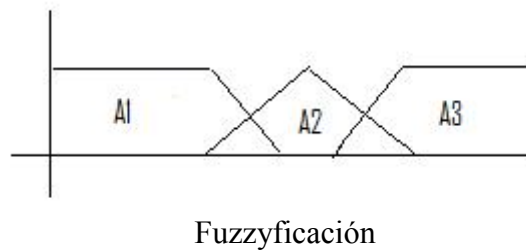
Esa función de pertenencia marcará si un valor pertenece totalmente a un conjunto, no pertenece o bien pertenece más o menos a ese conjunto (figura 5). Esos valores intermedios son los que marcan las diferencias respecto a los conjuntos clásicos en los que las funciones de pertenencia devuelven los valores uno o cero, dependiendo de si el valor pertenece o no al conjunto.

La función de pertenencia puede venir representada con formas prácticas: trapezoidal o triangular. Y también puede representarse con formas más teóricas como de campana, sinusoidal o irregulares.

El **control de inferencia** difuso está compuesto por tres grandes bloques:

1. fuzzyficación
2. Motor de Inferencia
3. defuzzyficación

La **fuzzyficación** es la encargada de tratar la información de entrada en conceptos fuzzy. En concreto transforma una medida del entorno a una representación en forma de función de posibilidad.



El **motor de inferencia** está formado por un conjunto de reglas lingüísticas de control que tienen la forma:

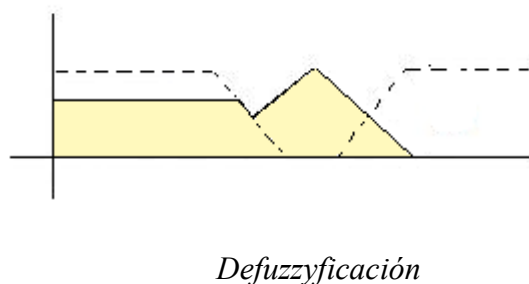
1. X es A1 ^ Y es B1 -> Z es C1
2. X es A2 ^ Y es B2 -> Z es C2

Donde X e Y serían unas variables de entrada, A1, A2 conjuntos borrosos de X, B1, B2 conjuntos borrosos de Y. Z sería una variable de salida, y C1, C2 conjuntos borrosos de Z.

El motor de inferencia si tuviera como entrada un hecho A' y otro B' daría como resultado C'. Obsérvese que el hecho A' o el B' no tiene porque pertenecer a un solo conjunto, puede pertenecer a varios, activando varias reglas. Al activar varias reglas, obviamente el resultado C' también estará formado por varios conjuntos.

Hay varias métodos para hallar ese conjunto final C'. Todos ellos dependen de como se interprete la intersección, disyunción o negación de los elementos que componen los antecedentes de las reglas. Y también de como se halle la implicación de las reglas.

La **defuzzyficación** es el proceso inverso a la fuzzyficación. Transforma el conjunto fuzzy final (figura 7) en un valor numérico, el método de centro de gravedad es el más conocido. O bien lo puede transformar en un valor cualitativo usando el método de la distancia, o el método de sumatorio cuenta.



Para este trabajo se ha usado la herramienta Fuzzy Toolbox de Matlab para realizar todos los cálculos fuzzy.

2.2.2. Controlador Fuzzy

La historia de controladores difusos tiene su origen en Mandami (Mandami, 1974) y Sugeno (Sugeno, 1985; Sugeno, 1995) que mediante una jerarquía de módulos básicos, las estrategias

borrosas planteadas posibilitan la ejecución de maniobras complejas. A partir de ahí, se han sucedido numerosas aplicaciones con esta técnica, haciendo acciones más agresivas sin perder la robustez y suavidad de la técnica.

La idea principal es crear un sistema que teniendo como entradas ciertas trayectorias deseadas, obtiene como salida la correspondiente consigna de control.

Si bien es cierto que no existe un método específico para modelar un controlador fuzzy, si que se pueden seguir una serie de directrices que son comunes para todos los controladores. Estos elementos o acciones son:

- Dominio de Entrada
- Dominio de Salida
- Definir Variables
- Definir etiquetas para variables
- Elemento de defuzzyficación
- Método de Inferencia

El Dominio de Entrada es el universo de valores de entrada que es permitido. Es decir, en nuestro caso serían las limitaciones propias del quadrotor a la hora de realizar los movimientos a controlar.

Igualmente el Dominio de Salida es el universo de acciones de control que es permitido. También está limitado en nuestro caso a las condiciones del vehículo.

Luego se definen las variables y el número de variables que representarán los dominios de entrada y salida. También es importante definir etiquetas (nombres) sencillos y significativos que representen a esas variables.

Finalmente se propone el elemento de defuzzyficación y el método de inferencia que se crea más apropiado para controlar el sistema. En caso de duda, métodos comunes y sencillos suelen ser buenas elecciones.

2.3. Planta del quadrotor.

Cuando se trata de modelar matemáticamente un quadrotor el primer paso es elegir una estructura general que pueda reflejar la complejidad dinámica de este tipo de vehículos aéreos. Deben influir dos factores fundamentalmente: la fiabilidad y la complejidad de las ecuaciones adaptadas. El paso complicado viene después, la obtención de los valores óptimos de los parámetros de la estructura.

2.3.1.Descripción

Un quadrotor, como ya se ha comentado, es un helicóptero formado por cuatro rotores. La disposición de los rotores es la de la figura 8. Es decir, tiene un rotor en la parte de delante, otro detrás, otro en la izquierda y finalmente otro en la derecha en posición simétrica.



quadrotor

El quadrotor que se quiere controlar posee una masa de quinientos gramos en la unidad central. Y una masa de cincuenta gramos en el conjunto motor-engranaje-rotor. La masa de las varillas se considerará despreciable.

El tamaño del vehículo es de seis centímetros de radio en la unidad central y de veintitrés centímetros desde cualquiera de los motores al centro del quadrotor. Es decir, una envergadura de cuarenta y seis centímetros.

2.3.2. Cinemática

Un quadrotor se controla variando únicamente la velocidad angular de los cuatro motores. En el motor delantero y trasero rotan en sentido horario mientras que los laterales rotan en sentido contrario a las manecillas del reloj. Con ello, los efectos giroscópicos y los momentos aerodinámicos tienden a cancelarse en vuelos estacionarios.

El movimiento de un cuerpo rígido posee seis grados de libertad: tres de ellos definen la posición de un punto de referencia en el cuerpo (normalmente el centro de masa, tal como será nuestro caso), y los otros tres definen la orientación del cuerpo.

$\mathbf{x} = (x, y, z)$ es el vector de posición del quadrotor.

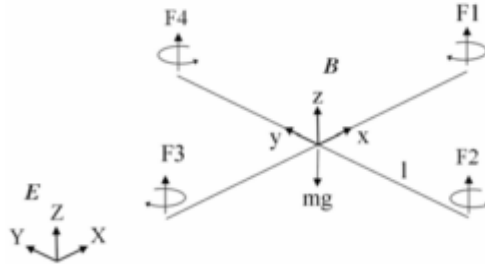
$\mathbf{v} = (u, v, w)$ es el vector de velocidad del quadrotor.

$\boldsymbol{\alpha} = (\phi, \theta, \Psi)$ son los ángulos de Euler del quadrotor (pitch, roll y yaw respectivamente).

$\boldsymbol{\omega} = (p, q, r)$ son las tasas de cambio de los ángulos.

La entrada de control principal o fuerza principal es la suma de las fuerzas producidas por cada motor de cada , esto es:

$$\mu = \sum_{i=1}^4 f_i \quad (1)$$



Sistema de Referencia del quadrotor

Para pasar del vector de posición al vector de velocidad es necesario utilizar la matriz de Euler.

$$R_{v/i}(\varphi, \theta, \psi) = \begin{pmatrix} 1 & 0 & 0 \\ 0 & \cos(\phi) & \sin(\phi) \\ 0 & -\sin(\phi) & \cos(\phi) \end{pmatrix} \begin{pmatrix} \cos(\theta) & 0 & -\sin(\theta) \\ 0 & 1 & 0 \\ \sin(\theta) & 0 & \cos(\theta) \end{pmatrix} \begin{pmatrix} \cos(\psi) & \sin(\psi) & 0 \\ \sin(\psi) & \cos(\psi) & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

Donde despejando se llega a la expresión:

$$\frac{d}{dt} \begin{pmatrix} x \\ y \\ z \end{pmatrix} = \begin{pmatrix} \dot{x} \\ \dot{y} \\ \dot{z} \end{pmatrix} = R_{v/i}(\varphi, \theta, \psi) \begin{pmatrix} u \\ v \\ w \end{pmatrix} \quad (2)$$

Y para relacionar $\boldsymbol{\omega}$ con $\frac{d}{dt} \boldsymbol{\alpha}$:

$$\frac{d}{dt} \begin{pmatrix} \varphi \\ \theta \\ \psi \end{pmatrix} = \begin{pmatrix} 1 & \sin(\phi) \tan(\theta) & \cos(\phi) \tan(\theta) \\ 0 & \cos(\phi) & -\sin(\phi) \\ 0 & -\sin(\phi) \operatorname{arccos}(\theta) & \cos(\phi) \operatorname{arccos}(\theta) \end{pmatrix} \begin{pmatrix} p \\ q \\ r \end{pmatrix} \quad (3)$$

2.3.3. Dinámica

Las aceleraciones en las coordenadas X, Y y Z son términos importantes a controlar. Hay que saber que la aceleración se puede extraer de la segunda ley de Newton; en la que se relacionan las fuerzas que actúan sobre el cuerpo con la masa total y la aceleración del quadrotor respecto del sistema de referencias. Puesto que la masa es un dato conocido, bastará con hallar las fuerzas que actúan para saber las aceleraciones.

Si analizamos las fuerzas a las que estará sometido el quadrotor, éstas serán la sustentación producida por cada motor, la ejercida por la gravedad, y los correspondientes rozamientos.

$$F_T = F_m - P - F_r \quad (4)$$

La fuerza del motor que tiene que expresar el movimiento en vertical viene dado por la siguiente expresión:

$$F_m = (0, 0, F_1 + F_2 + F_3 + F_4) \quad (5)$$

Desde el punto de vista inercial la fuerza queda de la siguiente forma:

$$F = \begin{pmatrix} \cos(\phi) \cdot \sin(\theta) \cdot \cos(\Psi) + \sin(\phi) \cdot \cos(\Psi) \cdot \sum_{i=1}^4 F_i \\ \cos(\phi) \cdot \sin(\theta) \cdot \cos(\Psi) - \sin(\phi) \cdot \cos(\Psi) \cdot \sum_{i=1}^4 F_i \\ \cos(\phi) \cdot \cos(\theta) \cdot \sum_{i=1}^4 F_i \end{pmatrix} \quad (6)$$

La fuerza del peso P viene dado por la expresión $P = m \cdot g$

Donde la gravedad se ha considerado constante e igual a 9.8 m/s^2

La fuerza de rozamiento ρ que sufre el vehículo, en cada eje de coordenadas, tiene la misma expresión:

$$F_r = \frac{1}{2} C_i \cdot A_c \cdot \rho \cdot i \cdot |i| \quad (7)$$

Donde: $i = x, y, z$.

El coeficiente C_i depende de varios factores (Stevens, 2003), pero para nuestro caso nos bastará considerarlo constante e igual a 1.3 en todos los casos.

Igualmente la densidad del aire ρ , la consideramos constante e igual a 1.293 Kg/m^3 .

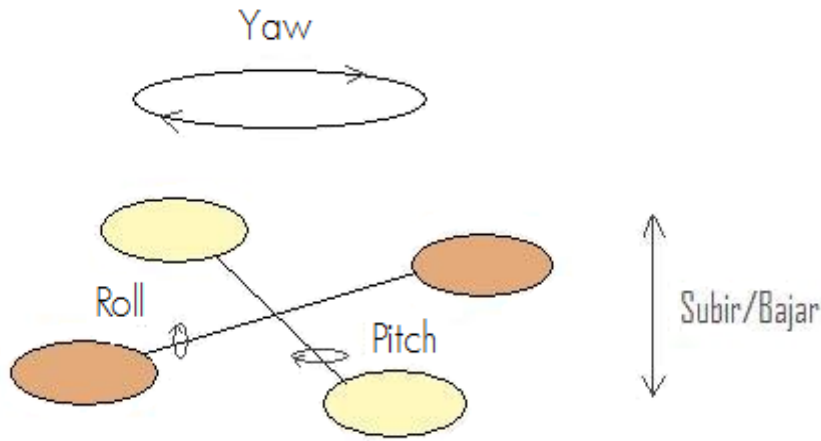
A_c es el área que presenta en su movimiento la unidad central (supuestamente esférica), por lo que

$$A_c = \pi \cdot R^2 = 0.0113 \text{ m}^2 \quad (8)$$

Como vemos, los términos correspondientes a estos rozamientos son muy pequeños, por lo que, aunque se han incorporado en la dinámica del quadrotor, podrían ser despreciados sin grandes efectos en el comportamiento del modelo simulado.

2.3.4. Movimientos Rotacionales

El movimiento general de un cuerpo rígido en el espacio se puede representar por la combinación de los movimientos de rotación y traslación.



Movimientos del quadrotor

Para el movimiento rotacional del quadrotor tendremos que:

$$dM / dt_i = T \quad (9)$$

Donde M es el momento angular y $T = (\tau_\phi, \tau_\theta, \tau_\psi)$ el torque aplicado.

Por una parte, $M = J \cdot \omega$ donde:

$$J = \begin{pmatrix} J_x & -J_{xy} & -J_{xz} \\ -J_{yx} & J_y & -J_{yz} \\ -J_{zx} & -J_{zy} & J_z \end{pmatrix} \quad (10)$$

Donde J es la matriz de inercia, que si consideramos al quadrotor como un elemento simétrico respecto de sus ejes, tomará la forma diagonal:

$$J = \begin{pmatrix} J_x & 0 & 0 \\ 0 & J_y & 0 \\ 0 & 0 & J_z \end{pmatrix} \quad (11)$$

Considerando al quadrotor como una esfera central de masa M y radio R de la que se proyectan cuatro varillas sin masa, de longitud l , en cuyos extremos se encuentra una masa puntual m , que representa al conjunto del motor y hélice, tendremos que:

$$J_i = \frac{2}{5} \cdot M \cdot R^2 + 2 \cdot l^2 \cdot m \quad \text{para } i = x, y \quad (12)$$

y

$$J_z = \frac{2}{5} \cdot M \cdot R^2 + 4 \cdot l^2 \cdot m \quad (13)$$

Por otra parte, los torques aplicados ($\tau_\phi, \tau_\theta, \tau_\psi$) tomarán distintas formas, al depender cada uno de una configuración distinta de los motores.

La rotación en torno al eje X (roll) se deberá a una diferencia entre las fuerzas F_2 y F_4 . Concretamente tendremos que:

$$\tau_\phi = l \cdot (F_2 - F_4) \quad (14)$$

Donde l es la distancia del centro del quadrotor a cualquiera de los rotores.

La rotación en torno al eje Y (pitch) se deberá a una diferencia entre las fuerzas F_1 y F_3 . Concretamente tendremos que:

$$\tau_\theta = l \cdot (F_1 - F_3) \quad (15)$$

La rotación del quadrotor entorno al eje Z (yaw) se deberá a una diferencia en el torque aplicado por los rotores delantera y trasero, y los laterales:

$$\tau_\psi = \tau_1 + \tau_2 - \tau_3 - \tau_4 \quad (16)$$

y tomando la ecuación de Coriolis:

$$dM/dt_i = dM/dt_v + \omega \times M = T \quad (17)$$

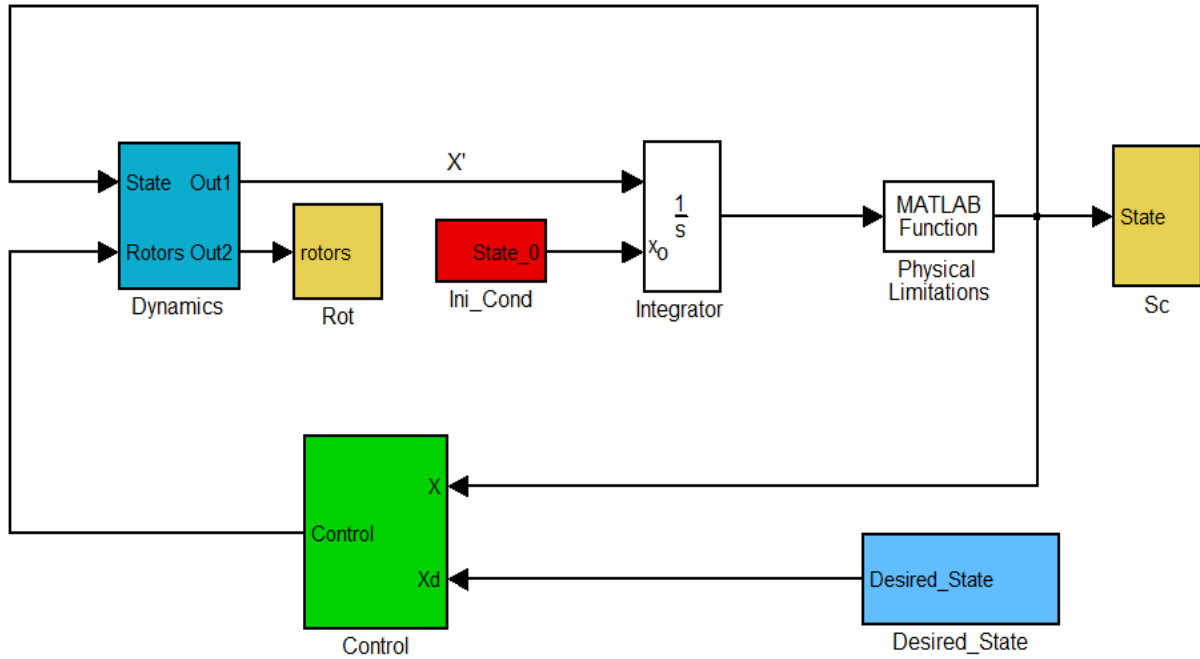
De esta manera llegamos a las conocidas ecuaciones de Euler que describen como las componentes del vector velocidad angular evolucionan en el tiempo, en respuesta a los torques aplicados:

$$d(J \cdot \omega)/dt_v + \omega \times (J \cdot \omega) = (\tau_\phi, \tau_\theta, \tau_\psi) \quad (18)$$

$$J \begin{pmatrix} \dot{p} \\ \dot{q} \\ \dot{r} \end{pmatrix} + \begin{pmatrix} i & j & k \\ p & q & r \\ J_x p & J_y q & J_z r \end{pmatrix} = \begin{pmatrix} \tau_\phi \\ \tau_\theta \\ \tau_\psi \end{pmatrix} \quad (19)$$

$$\begin{pmatrix} \dot{p} \\ \dot{q} \\ \dot{r} \end{pmatrix} = J^{-1} \cdot \left\{ \begin{pmatrix} \tau_\phi \\ \tau_\theta \\ \tau_\psi \end{pmatrix} - \begin{pmatrix} i & j & k \\ p & q & r \\ J_x p & J_y q & J_z r \end{pmatrix} \right\} = \begin{pmatrix} \frac{\tau_\phi}{J_x} \\ \frac{\tau_\theta}{J_x} \\ \frac{\tau_\psi}{J_x} \end{pmatrix} + \begin{pmatrix} \frac{J_x - J_z}{J_x} \cdot q \cdot r \\ \frac{J_z - J_x}{J_y} \cdot p \cdot r \\ \frac{J_x - J_y}{J_z} \cdot p \cdot q \end{pmatrix} \quad (20)$$

2.4. Descripción del Modelo



Modelo del quadrotor

El modelo (Sanchez,2009) es el que se muestra en la figura 11, consta de los siguientes bloques:

- Dynamics
- Rotors
- Ini_Cond
- Integrator
- Physical Limitations
- Sc
- Control
- Desired State

Veamoslos todos haciendo hincapié en los dos módulos más importantes: Dynamics y Control.

2.4.1. Dynamics

Entrada: State, Rotors

Salida: X', Rotor

Ficheros: quad_dynamics.m, limitations.m

Este bloque es la **planta** del sistema, contiene las ecuaciones físicas que describen su comportamiento. Recibe como entrada el estado actual del quadrotor y la potencia necesaria que se debe ejercer a cada rotor para llegar al estado que se desea. Con esas entradas se halla el estado siguiente que tomará en el próximo período de tiempo. Obsérvese que sólo es el siguiente estado, podría coincidir o no con el estado deseado. Con el tiempo el sistema se estabiliza y el estado actual y el estado siguiente coincidirán con el estado deseado.

Estas ecuaciones físicas están guardadas en el archivo de Matlab quad_dynamics.m, el cual se apoya en el fichero auxiliar limitations.m que contiene datos constantes relativos al quadrotor como características del quadrotor y suposiciones físicas-medioambientales.

2.4.2. Rotor

Entrada: Rotor

Bloque sencillo que muestra por pantalla la suma de la potencia de los rotores. Es útil porque es un parámetro fundamental para calcular las posiciones geométricas del vehículo.

2.4.3. Ini_Cond

Salida: State_0

En este bloque se fijan las condiciones iniciales del quadrotor, es decir, define el estado inicial. Es de gran utilidad para hacer diferentes pruebas.

2.4.4. Integrator

Entrada: X', X0

Salida: St

El integrator es una función integral del Simulink de Matlab que sirve para mejorar la sintonía y eliminar ruidos. La primera vez recibirá como entra X0 con el estado inicial del quadrotor, luego cuando el sistema se realimente en sucesivas iteraciones recibirá como entrada X'. Devuelve el estado refinado que le llegue a la entrada.

2.4.5. Physical Limitations

Entrada: St

Salida: State

Ficheros: physical_limitations.m

A este módulo le llega el estado actual y se encarga de restringir valores de estado que no podemos permitir.

Entre las limitaciones están:

- I. No permitir alturas negativas. Lo que significaría que no se permitiría que el quadrotor

estuviera por debajo del nivel del suelo. En caso de que a este módulo le llegará un valor negativo lo pondría a cero y todas los ángulos y velocidades se pondrían a cero también.

- II. Limitaciones en los ángulos y en la velocidad del quadrotor son necesarias para garantizar la estabilidad del vehículo. Esto es útil cuando el controlador produce señales de control muy bruscas ante errores muy grandes. Los límites marcarían de esta forma el límite de seguridad. Es el caso del roll y del pitch que sólo se permitirán giros en el rango de (-1.3 , 1.3) radianes.

2.4.6. Sc

Entrada: State

En este módulo se muestra la respuesta del sistema, es decir muestro el estado actual del modelo.

2.4.7. Control

Entrada: X, Xd

Salida: Rotors

Ficheros: seleccion.m, controlZPID.fis, controlPitchPID.fis, controlRollPID.fis, controlYawPID.fis

Es el módulo del **controlador** del sistema, contiene los archivos **fuzzy** para estabilizar el sistema. Recibe como entradas el estado actual y el estado deseado. Selecciona de ahí las variables necesarias para el controlador fuzzy a través del fichero seleccion.m y con esa información el controlador es capaz de extraer las señales de control necesarias para estabilizar el sistema, que serán enviadas al bloque dynamics. El controlador está formado por cuatro ficheros hechos con la herramienta ToolboxFuzzyLogic: controlZPID.fis, controlPitchPID.fis y controlRollPID.fis, controlYawPID.fis. En el apartado ControladorFuzzy se explicará con mayor detenimiento.

2.4.8. Desired State

Salida: Desired_State

Este bloque es el equivalente al Ini_Cond, su función es fijar los valores de estado deseados. Se usa junto al Ini_Cond para probar el sistema.

Controlador Fuzzy

En este apartado se verá con detenimiento el controlador que estabiliza el sistema ubicado en el bloque Control.

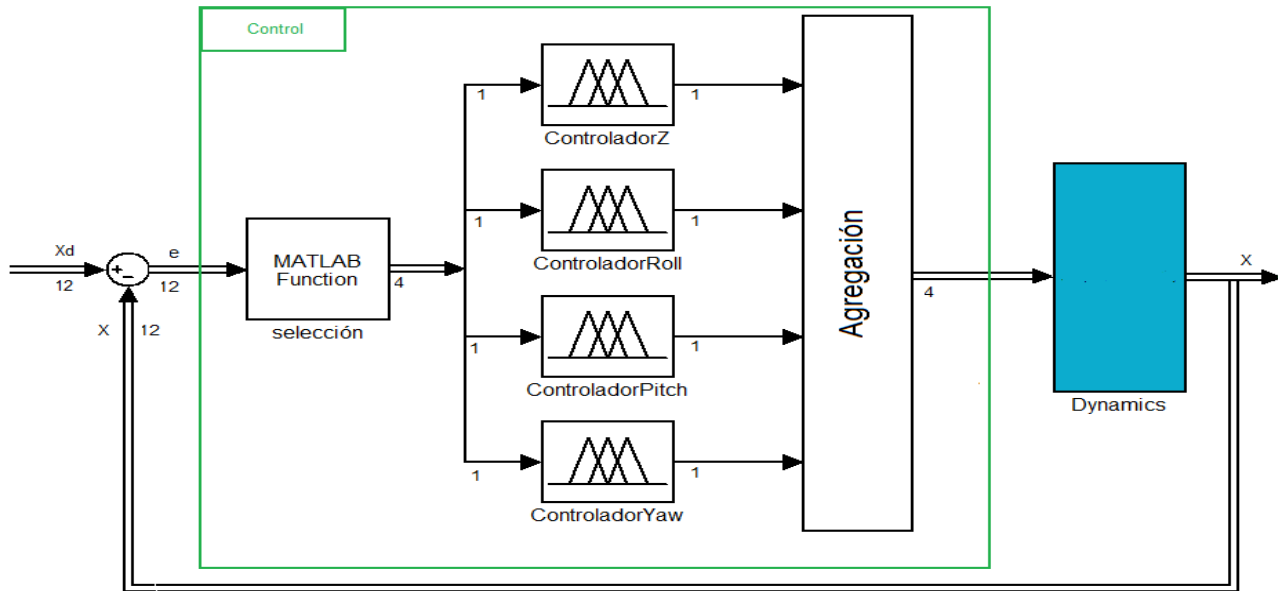


Diagrama del bloque Control.

En control se recibe como entrada un bus del error de todas las variables, doce en total. Previamente se ha tenido que calcular el error, restando al estado deseado el estado actual ($X_d - X$).

La salida del bloque Control es un vector de cuatro elementos correspondientes a la potencia de cada uno de los motores. Ese vector va a la entrada de la planta (Dynamics) y la salida es el estado actual X del sistema que se realimenta como se ve en la figura 1.

El primer subbloque es el de **Selección**, contiene el fichero seleccion.m que filtra de las doce variables posibles las que se desean aplicar en los controladores fuzzy. En esta versión final están habilitadas cuatro: Z, Roll, Pitch y Yaw. Además este subbloque ha sido a lo largo del desarrollo un mecanismo sencillo para deshabilitar o habilitar las variables según las necesidades.

Existen cuatro controladores, uno por movimiento: controladorZ, controladorRoll, controladorPitch y controladorYaw, en cuyo interior están los dispositivos fuzzy que forman la lógica de control (figura 1). La salida de control es la potencia de motor, que va al módulo de Agregación cuyo objetivo es administrar la potencia necesaria a cada uno de los motores.

Veamos ahora primeramente el módulo de Agregación y luego nos centraremos en los controladores.

3.1. Módulo Agregación

Agregación es un módulo intermediario entre los controladores y la planta del sistema. Su funcionalidad es liberar de tarea a los controladores fuzzy reduciendo el número de valores de salida de estos y simplificando las labores de control. La reducción consiste en bajar de los cuatro

posibles valores de salida, uno por cada motor, a un valor de salida (Figura 1). Por tanto Agregación se encargará de hallar la potencia final de los cuatro motores a partir de la potencia que cada controlador cree necesaria para ejecutar su movimiento.

Veamos de manera independiente como interpreta los valores que le llegan de cada uno de los controladores.

El controladorZ devuelve la potencia necesaria de cada rotor para corregir el error de la altura. Recordemos que la potencia de cada rotor para subir o bajar es la misma para cada uno, por eso es suficiente que el fuzzy sólo produzca la salida de un rotor y luego usando un demultiplexor añadir a los cuatro buses de los rotores.

El controladorRoll, controladorPitch y el controladorYaw operan de manera semejante, en el sentido es que los tres devuelven incrementos de la potencia de los motores que se le deberá sumar a la potencia de los motores necesario para estabilizar la altura. De tal forma que el controlador de la altura es la base sobre la que luego se añaden los incrementos del resto de controladores.

De manera particular:

ControladorRoll: Actúa sobre los motores M2 y M4, es decir sobre los rotores laterales. Ya que el giro es sobre el eje X, bastará con mantener los rotores delantero y trasero (M1 y M3) y variar los rotores laterales. Además, todo lo que se incremente de un motor, se debe decrementar del otro, para que se conserve la fuerza μ constante y no se desestabilice el quadrotor.

De esta forma el fuzzy sólo debe devolver un único valor: el incremento en valor absoluto de los motores laterales M2 y M4 (incRoll). Luego, bastará sumarlo como incremento de M2 y como incremento de M4 a lo devuelto por el ControladorZ con signos opuestos.

Para justificar el signo es necesario saber que el ControladorRoll devuelve una potencia positiva si se quiere girar en sentido horario (hacia la derecha) y una potencia negativa si se quiere girar en sentido antihorario (hacia la izquierda). Por otra parte, al girar a la derecha es necesario que el motor izquierdo (M2) tenga más potencia que el derecho (M4); y viceversa, al girar a la izquierda, que el motor derecho (M4) tenga más potencia que el izquierdo (M2). Luego la expresión resultante queda:

$$\begin{aligned} M2 &= M2_{\text{antRoll}} + \text{incRoll} \\ M4 &= M4_{\text{antRoll}} - \text{incRoll} \end{aligned}$$

Así:

- si $\text{incRoll} > 0$ entonces $M2 > M4$. OK
- si $\text{incRoll} < 0$ entonces $M4 > M2$. OK

ControladorPitch: Actúa sobre los motores M1 y M3, es decir sobre el rotor delantero y el trasero. Ya que el giro es sobre el eje Y, bastará con mantener los rotores laterales (M2 y M4) y variar los rotores delantero y trasero. Además, al igual que en el roll, todo lo que se incremente de un motor, se debe decrementar del otro, para que se conserve la fuerza μ constante y no se desestabilice el quadrotor.

De esta forma el fuzzy sólo debe devolver un único valor: el incremento en valor absoluto de los motores M1 y M3 (incPitch). Luego, bastará sumarlo como incremento de M1 y como incremento

de M3 a lo devuelto por el ControladorZ con signos opuestos.

Para justificar el signo es necesario saber que el ControladorPitch devuelve una potencia positiva si se quiere girar en sentido horario (girar hacia abajo) y una potencia negativa si se quiere girar en sentido antihorario (girar hacia arriba). Por otra parte, al girar hacia abajo es necesario que el motor trasero (M3) tenga más potencia que el frontal (M1); y viceversa, al girar hacia arriba, que el motor delantero (M1) tenga más potencia que el trasero (M3). Luego la expresión resultante queda:

$$M1 = M1_{antPitch} - incPitch$$

$$M3 = M3_{antPitch} + incPitch$$

Así:

- si $incPitch > 0$ entonces $M3 > M1$. OK
- si $incPitch < 0$ entonces $M1 > M3$. OK

ControladorYaw: Actúa sobre los motores M1, M2, M3 y M4. El giro es sobre el eje z y es necesario modificar todos los rotores. Sabemos que para que se pueda producir el yaw los rotores se agrupan por pares: los laterales (M2 y M4) girar en un sentido y el delantero y trasero (M1 y M3) en el contrario. Además, para que se conserve la fuerza u constante todo lo que se incremente / decremente de los motores laterales, se debe hacer lo opuesto en los otros otros.

De esta forma el fuzzy sólo debe devolver un único valor: el incremento en valor absoluto que tienen que tener los motores ($incYaw$). Luego, bastará sumarlo a lo devuelto por el ControladorZ como incremento de M2 y M4 por un lado y como incremento de M1 y M3 por otro lado, pero con signos opuestos.

Para justificar el signo es necesario saber que el ControladorYaw devuelve una potencia positiva si se quiere girar en sentido horario (hacia la derecha) y una potencia negativa si se quiere girar en sentido antihorario (hacia la izquierda). Por otra parte, al girar a la derecha es necesario que los motores delantero y trasero (M1 y M3) tengan más potencia que los laterales (M2 y M4); y viceversa, al girar a la izquierda, que los motores laterales (M2 y M4) tengan más potencia que los delantero y trasero (M1 y M3). Luego la expresión resultante queda:

$$M1 = M1_{antYaw} + incYaw$$

$$M2 = M2_{antYaw} - incYaw$$

$$M3 = M3_{antYaw} + incYaw$$

$$M4 = M4_{antYaw} - incYaw$$

Así:

- si $incYaw > 0$ entonces $M1, M3 > M2, M4$. OK
- si $incYaw < 0$ entonces $M2, M4 > M1, M3$. OK

Como observación final se puede afirmar que el controladorZ influye en el resto de controladores. El controladorRoll y controladorPitch tienen su influencia en el controladorYaw. Sin embargo, se ha conseguido con esta estrategia eliminar la dependencia entre el controladorRoll y el controladorPitch.

3.2. Control Fuzzy PID

Los controladores reciben cada uno el error de la variable que van a controlar y calculan la derivada del error y la integral del error. A cada una de las tres variables se les fija una ganancia (G_E , G_{DE} , G_{IE} respectivamente) y formarán las entradas de los dispositivos fuzzy que forman la lógica de control.

Para la lógica de control se ha utilizado la Fuzzy Logic Toolbox de Simulink y se han creado los ficheros controlZPID.fis, controlRollPID.fis, controlPitchPID.fis y controlYawPID.fis para el controlador de la altura, el del roll, el del pitch y el del yaw respectivamente.

A las salidas finalmente se le añaden unas ganancias (G_U) y entran al módulo de Agregación.

Los cuatro controladores tienen los siguientes puntos en común:

- Todas las variables de entrada y de salida tienen un número impar de funciones de pertenencia con forma triangular la central y trapezoidal el resto.
- Las reglas han seguido el siguiente formato: Si e_Z es Alto and dE es Positiva entonces U es Grande.
- Se ha utilizado la estructura de Mandani como tipo de inferencia.
- Se ha usado el centro de gravedad como método de fuzzyficación.
- Se ha utilizado para el método AND la función mínimo y para el método IMPLICATION la función mínimo también.

3.2.1. Control Fuzzy PID Altura

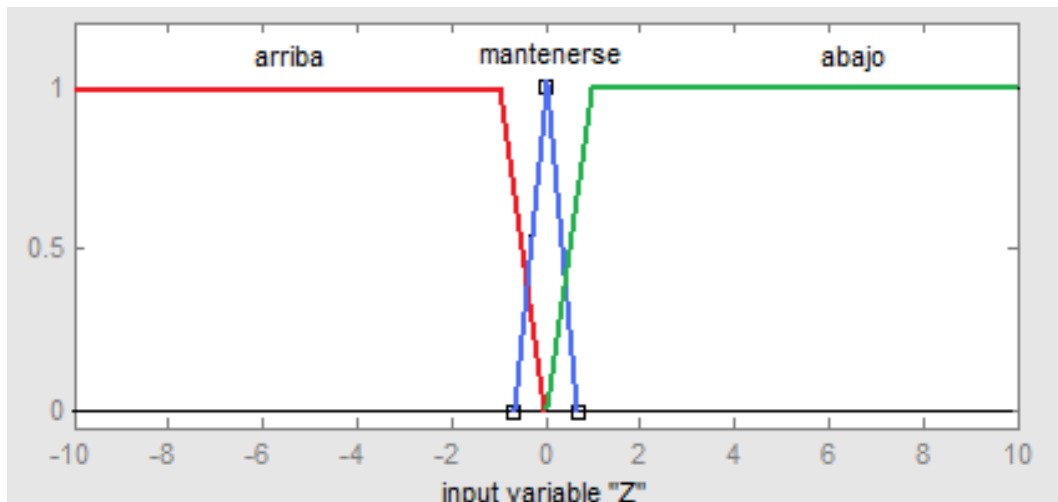
Es el encargado de hallar la potencia necesaria para controlar la altura del cuadrotor. Realmente para afinar los parámetros de este primer controlador solamente es necesario disponer de las ecuaciones relativas al movimiento vertical, excluyendo las relativas a los movimientos rotatorios.

Los datos de entrada son tres:

- Error de la altura (Z)
- Derivada del error de la altura (dZ)
- Integral del error de la altura (iZ)

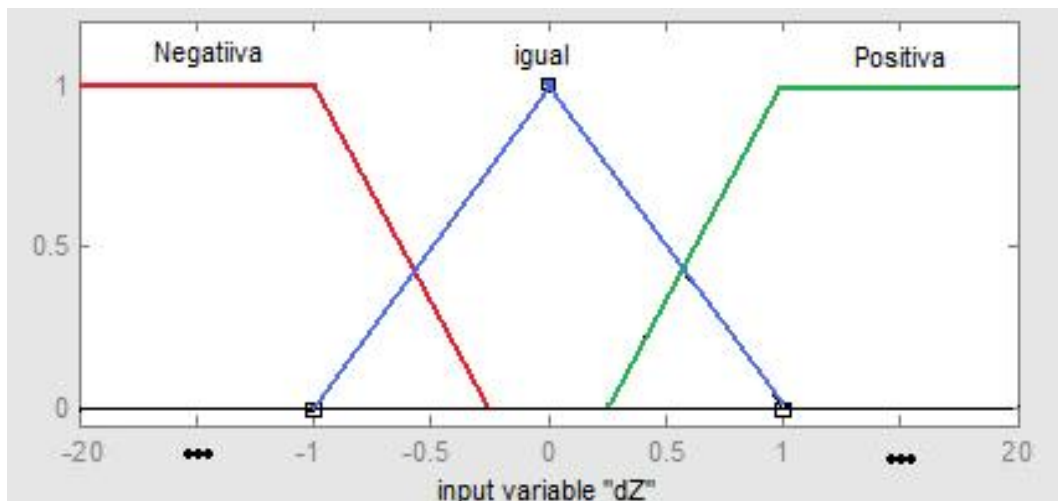
A continuación se incluyen tanto los conjuntos borrosos de las variables de entrada como los de la salida. Se proporcionan los valores de las ganancias y se añaden también una tabla con las acciones de control.

Para la variable de entrada correspondiente al error de la altura (Z) se ha dividido su universo en tres conjuntos borrosos: arriba, mantenerse y abajo.



Conjuntos Borrosos de la variable de entrada Z

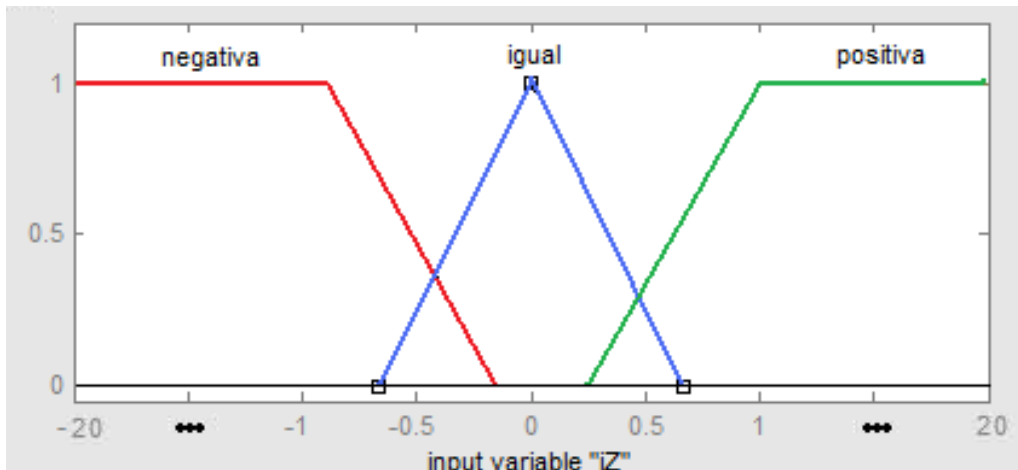
La variable de entrada correspondiente a la derivada del error de la altura (dZ) dispone de tres conjuntos borrosos: Negativa, igual, Positiva.



Conjuntos Borrosos de la variable de entrada de entrada dZ

La integral de la altura (iZ) tiene también tres conjuntos borrosos: Negativa, igual, Positiva.

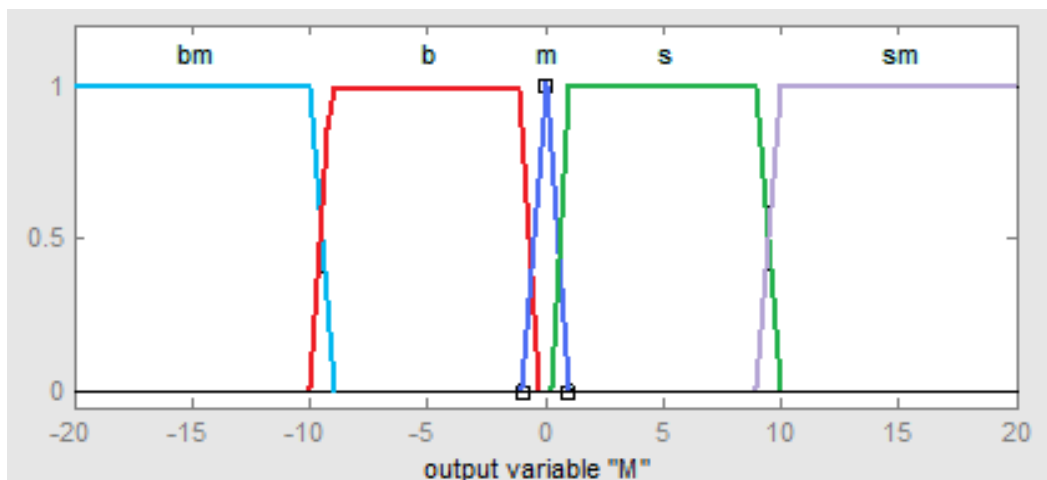
Aunque tengan nombres iguales no son los mismos conjuntos como se pueden ver en las funciones de pertenencia de la figura 3 y 4.



Conjuntos Borrosos de la variable de entrada de entrada iZ

La variable de salida que hace referencia a la potencia del motor, toma valores absolutos, esto será diferente al dominio de salida del resto de variables que son sólo incrementos de potencia y el rango de valores es más reducido.

Se han utilizado cinco conjuntos borrosos para la salida de control : BajarMucho (bm), Bajar (b), Mantenerse (m), Subir (s) y SubirMucho (sm). El método de defuzzificación es el de centro de gravedad.



Conjuntos Borrosos de la variable de entrada de salida M

Los valores de las ganancias ante ensayo y error han sido:

$$GE = 1$$

$$GDE = 0.5$$

$$GIE = 10$$

$$GU = 1$$

En la siguiente tabla se relaciona unequivocamente para las distintas combinaciones de entrada el conjunto de salida que le corresponde.

DeZ - IeZ \ eZ	Arriba	Mantenerse	Abajo
Negativa-Negativa	bm	b	m
Negativa-Igual	bm	b	m
Negativa-Positiva	bm	b	m
Igual-Negativa	b	s	s
Igual-Igual	b	m	s
Igual-Positiva	b	m	s
Positiva-Negativa	s	s	sm
Positiva-Igual	s	s	sm
Positiva-Positiva	s	s	sm

Tabla 1: Reglas de Inferencia de la altura

3.2.2. Control Fuzzy Roll

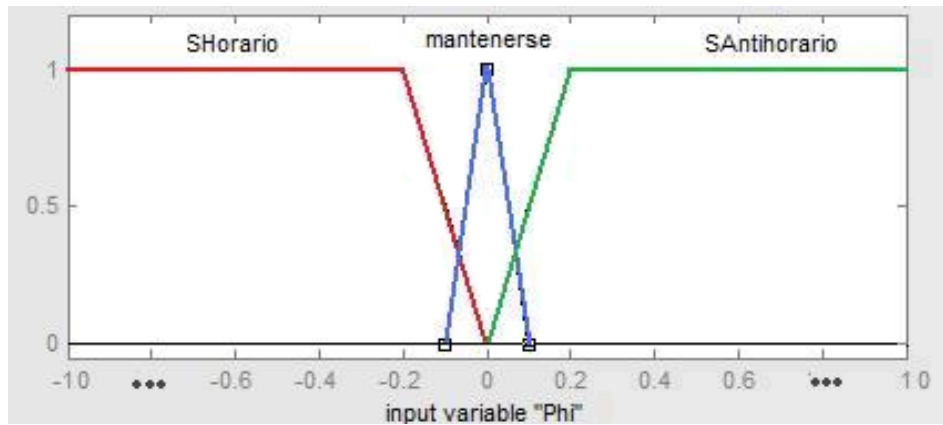
Para el control del roll se necesitan utilizar más ecuaciones físicas, pues ahora es necesario controlar los movimientos rotacionales y provocar respuestas en el desplazamiento lateral del quadrotor. También es necesario hacer las calibraciones oportunas equilibrando desajustes que pudieran afectar a la altura.

Los datos de entrada son tres:

- Error del roll (Φ)
- Derivada del error del roll ($d\Phi$)
- Integral del error del roll ($i\Phi$)

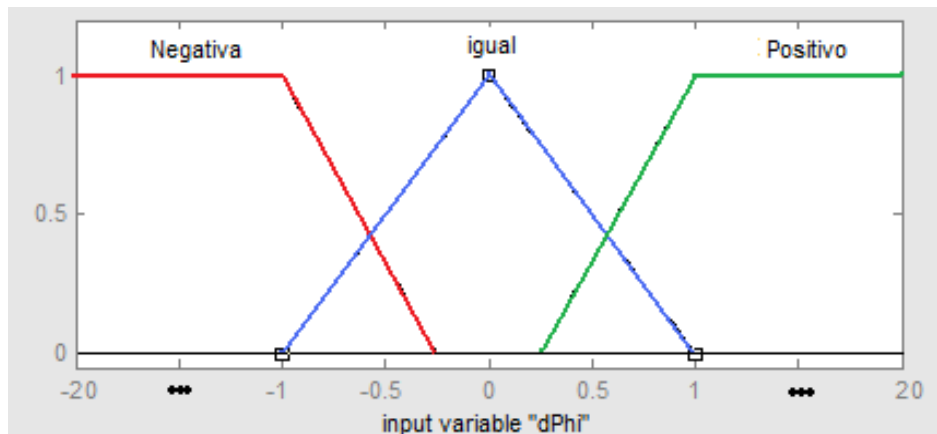
Se ha particionado el universo de discursos de las variables de entrada en tres conjuntos borrosos. Definidas las de los extremos con funciones de pertenencia trapezoidales y la central de forma triangular. A cada conjunto borroso le asociamos una etiqueta lingüística.

Los tres conjuntos de la variable del error (Φ) se han llamado Shorario, mantenerse, Santihorario. En referencia a si el giro que se quiere realizar es en sentido horario, en sentido antihorario o se quiere mantener el rumbo. El rango de valores para esta variable lingüística se muestra en la figura 6:



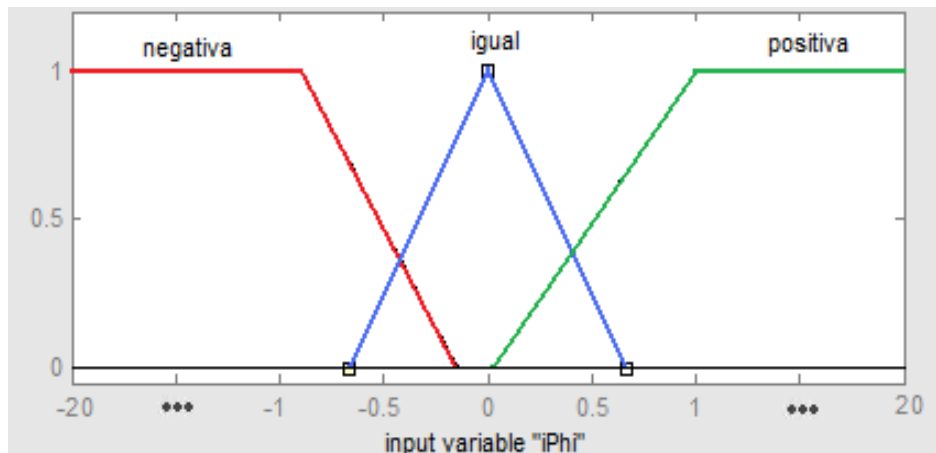
Conjuntos Borrosos de la variable Phi

Los tres conjuntos de la variable $d\Phi$ se han llamado Negativo, igual, Positivo. Etiquetas lingüísticas que recogen si el valor de la derivada del error del roll es positiva, negativa o bien es aproximadamente cero.



Conjuntos Borrosos de la variable de entrada dPhi

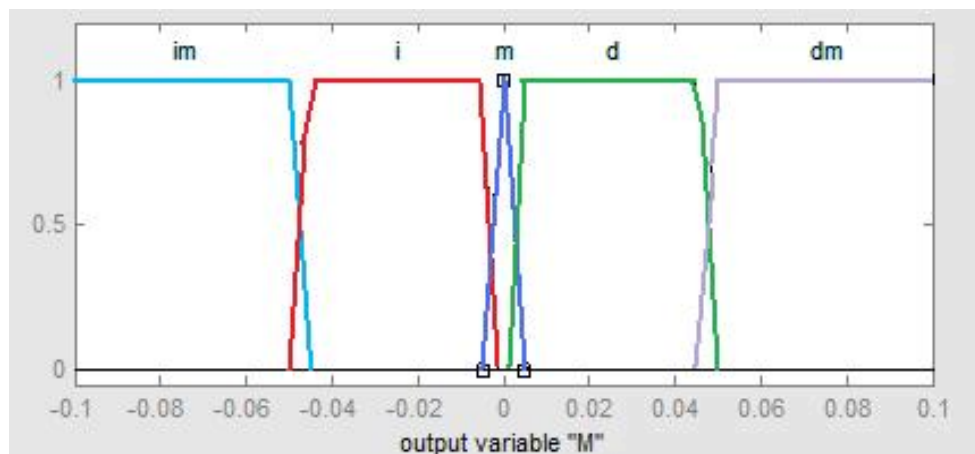
De manera análoga los tres conjuntos de la variable $d\Phi$ se han llamado Negativo, igual, Positivo. Etiquetas lingüísticas que recogen si el valor de la integral del error del roll es positiva, negativa o bien es aproximadamente cero.



Conjuntos Borrosos de la variable de entrada de entrada iPhi

Cuando se vio el modulo de Agregación ya se demostró que sólo es necesario una salida y luego ya se hacen los sencillos cálculos aritméticos que permiten saber el del resto.

A diferencia de las variables de entrada para el espacio de salida, escogemos cinco conjuntos borrosos. El número queda marcado por el número de reglas que se tienen en mente realizar. Las etiquetas de esta variable de control son im: izquierda mucho, i: izquierda, m: mantenerse, b: bajar, bm: bajar mucho. En la siguiente figura se muestra el rango de valores y sus particiones con las cinco funciones de pertenencia. El dominio de salida como ya se comentaba es menor pues representan incrementos de potencia a añadir.



Conjuntos Borrosos de la variable de salida M del roll

Los valores de las ganancias obtenidas ante ensayo y error han sido:

$$GE = 1$$

$$GDE = 3$$

$$GIE = 5$$

$$GU = 1$$

En la tabla 2 se muestra las acciones de control para los tres posibles valores de entrada mostrados. Aunque el número de reglas teóricas es 3^3 , se han podido agrupar algunas acciones en una sola regla ya que la acción de control es la misma reduciendo el número de reglas a once.

DePhi-IePhi \ ePhi	Shorario	Mantenerse	SAntihorario
Negativa-Negativa	dm	d	i
Negativa-Igual	dm	d	i
Negativa-Positiva	dm	d	i
Igual-Negativa	d	d	i
Igual-Igual	d	m	i
Igual-Positiva	d	i	i
Positiva-Negativa	d	i	im
Positiva-Igual	d	i	im
Positiva-Positiva	d	i	im

Tabla 2: Reglas de Inferencia del roll

3.2.3. Control Fuzzy Pitch

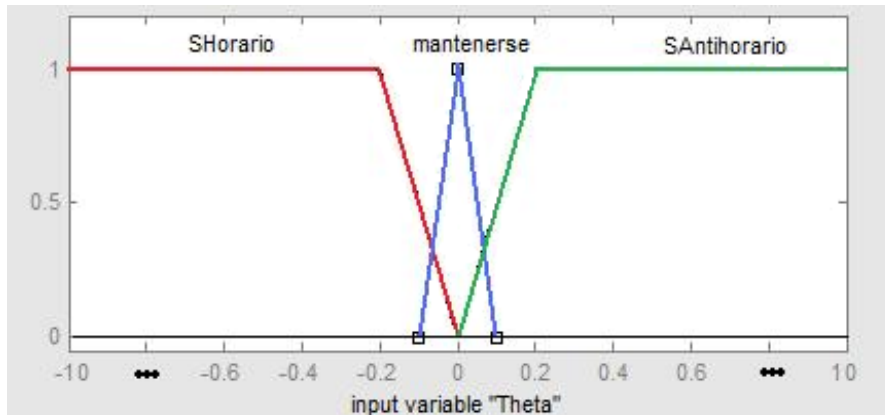
Para el control del pitch, al igual que en el roll, se necesitan nuevas ecuaciones físicas que modelicen el giro y respondan en el desplazamiento por el eje X del quadrotor. Este movimiento también afecta a la altura pero no al roll ya que las acciones de control actúan en motores diferentes como ya se ha comentado anteriormente.

Los datos de entrada que maneja el controlador del pitch son tres:

- Error del pitch (Theta)
- Derivada del error del pitch (dTheta)
- Integral del error del pitch (iTheta)

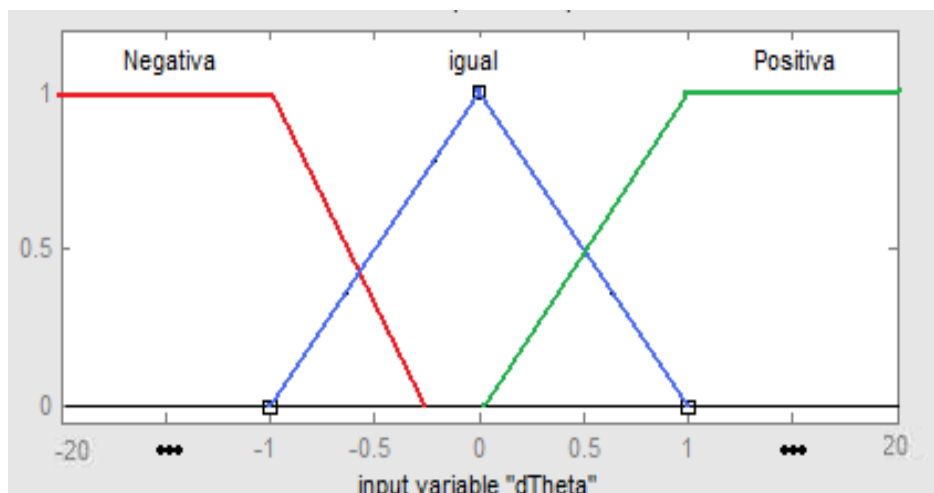
Las variables de entrada disponen cada una de tres conjuntos borrosos. Definidas las de los extremos con funciones de pertenencia trapezoidales y la central de forma triangular.

Los tres conjuntos de la variable del error (Theta) se han llamado Shorario, mantenerse, Santihorario. En referencia a si el giro en el eje Y que se quiere realizar es en sentido horario, en sentido antihorario o si quiere mantener el rumbo (ver figura 10).



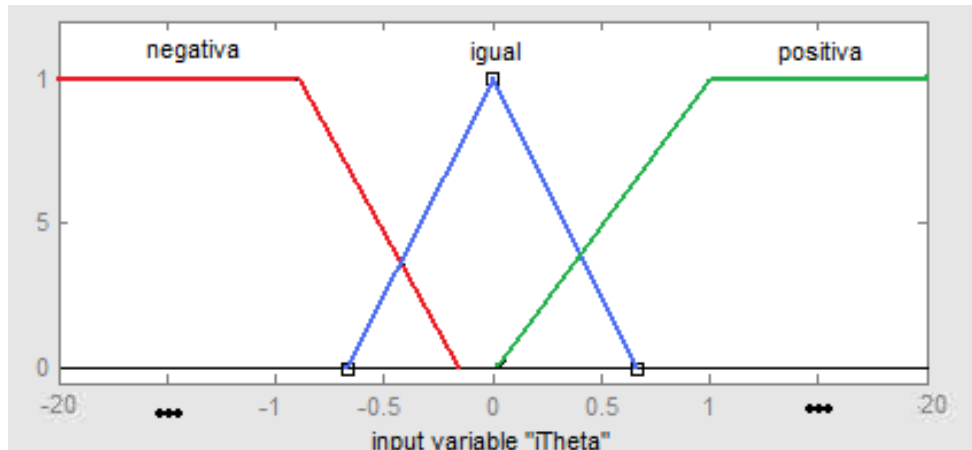
Conjuntos Borrosos de la variable de entrada Theta

Los tres conjuntos de la variable dTheta se han llamado Negativo, igual, Positivo. Etiquetas lingüísticas que recogen si el valor de la derivada del error del pitch es positiva, negativa o bien es aproximadamente cero.



Conjuntos Borrosos de la variable de entrada dTheta

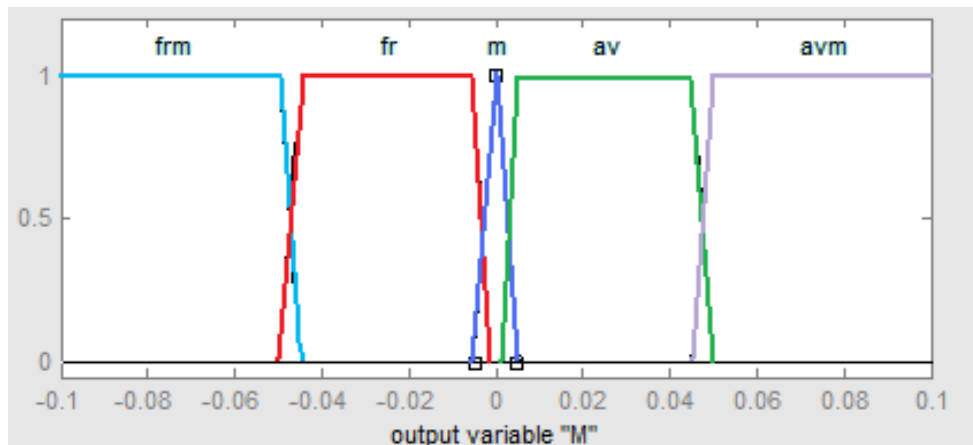
De manera análoga los tres conjuntos de la variable iTheta se han llamado Negativo, igual, Positivo. Etiquetas lingüísticas que recogen si el valor de la integral del error del pitch es positiva, negativa o bien es aproximadamente cero.



Conjuntos Borrosos de la variable de entrada iTheta

En el apartado en el que vimos el controlador ya demostramos en el apartado 6 que sólo es necesario una salida.

A diferencia de las variables de entrada para el espacio de salida, escogemos cinco conjuntos borrosos. Las etiquetas de esta variable de control son frm: frenar mucho, f: frenar, m: mantenerse, av: avanzar, avm: avanzar mucho. En la siguiente figura aparecen dichas particiones para el rango de valores $[-0.1, 0.1]$.



Conjuntos Borrosos de la variable de salida M para el pitch

Los valores de las ganancias obtenidas ante ensayo y error han sido:

$$GE = 1$$

$$GDE = 3$$

$$GIE = 5$$

$$GU = 1$$

En la tabla 3 se muestra las acciones de control para los tres posibles valores de entrada explicados. Al igual que en el roll el número de reglas es once porque se han agrupado las acciones de control equivalentes.

DeTheta-\dot{e}Theta \ eTheta	Shorario	Mantenerse	SAntihorario
Negativa-Negativa	avm	av	fr
Negativa-Igual	avm	av	fr
Negativa-Positiva	avm	av	fr
Igual-Negativa	av	av	fr
Igual-Igual	av	m	fr
Igual-Positiva	av	fr	fr
Positiva-Negativa	av	fr	frm
Positiva-Igual	av	fr	frm
Positiva-Positiva	av	fr	frm

Tabla 3: Reglas de Inferencia del pitch

3.2.4. Control Fuzzy Yaw

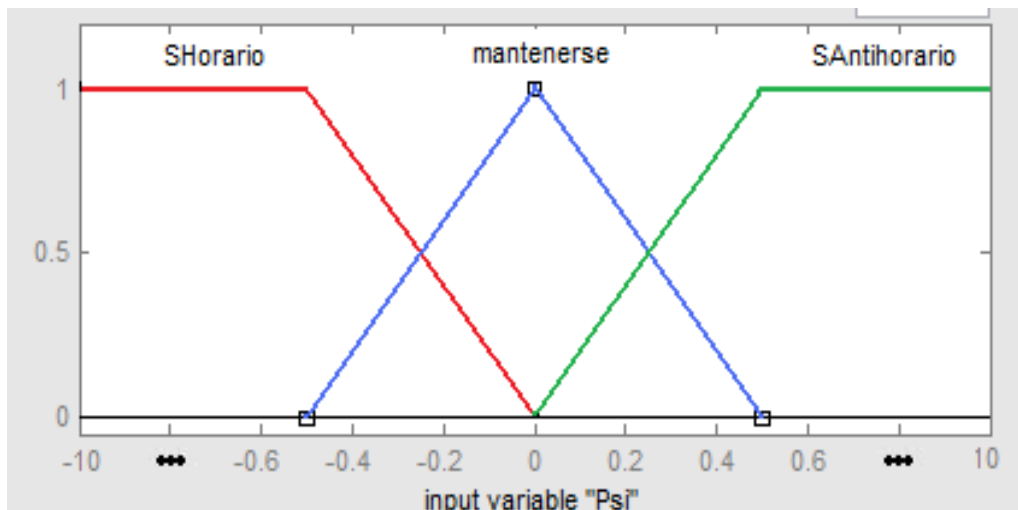
Para el control de el yaw se necesitan utilizar las ecuaciones físicas relativas a el yaw. También es necesario hacer las calibraciones oportunas equilibrando desajustes que pudieran afectar a la altura, al pitch y al roll.

Las variables de entrada como en los anteriores movimientos son tres:

- Error de el yaw (Psi)
- Derivada del error del yaw (dPsi)
- Integral del error del yaw (iPsi)

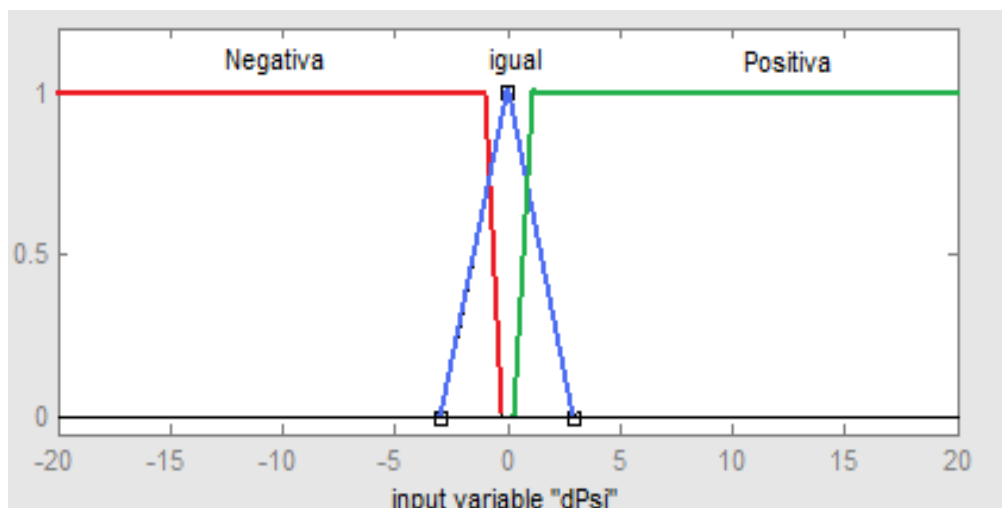
Se ha particionado el universo de discursos de las variables de entrada en tres conjuntos borrosos. Definidas las de los extremos con funciones de pertenencia trapezoidales y la central de forma triangular. A cada conjunto borroso le asociamos una etiqueta lingüística.

Los tres conjuntos de la variable del error (Psi) se han llamado Shorario, mantenerse, Santihorario. En referencia a si el giro que se quiere realizar es en sentido horario, en sentido antihorario o se quiere mantener el rumbo. El rango de valores para esta variable lingüística se muestra en la figura 14:



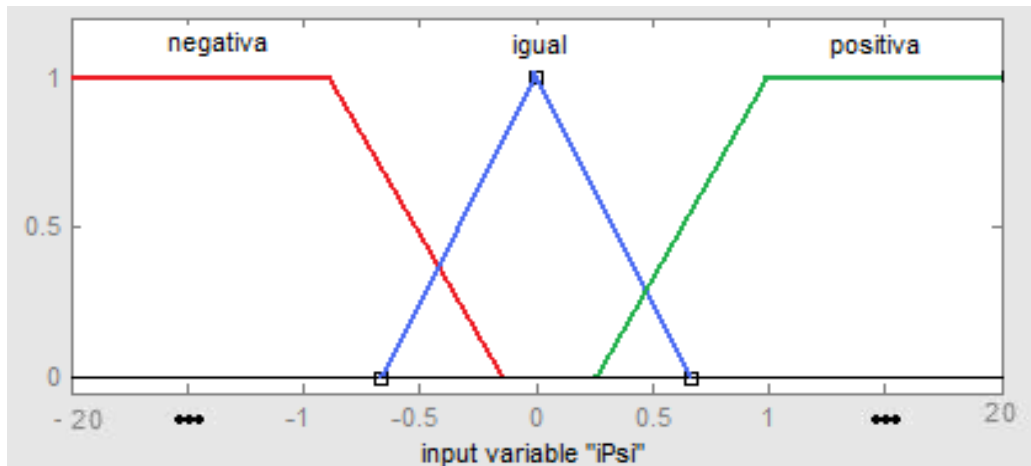
Conjuntos Borrosos de la variable de entrada Psi

Los tres conjuntos de la variable dPsi se han llamado Negativo, igual, Positivo. Etiquetas lingüísticas que recogen si el valor de la derivada del error del roll es positiva, negativa o bien es aproximadamente cero.



Conjuntos Borrosos de la variable de entrada dPsi

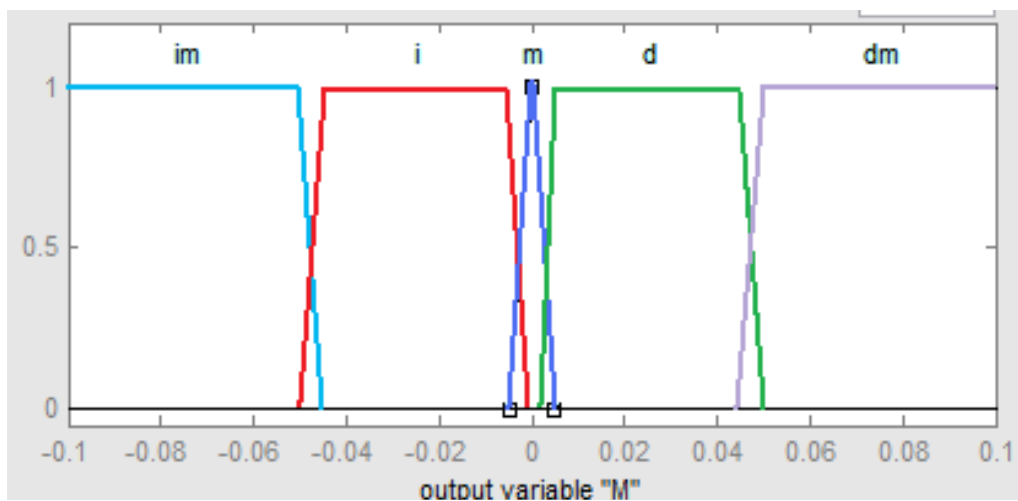
De manera análoga los tres conjuntos de la variable dPsi se han llamado Negativo, igual, Positivo. Etiquetas lingüísticas que recogen si el valor de la integral del error de el yaw es positiva, negativa o bien es aproximadamente cero.



Conjuntos Borrosos de la variable de entrada iPsi

Sólo hay una salida que es como en los otros dos movimientos el incremento del motor que se cree necesario, el intervalo de valores de salida es de $[-0.1, 0.1]$.

A diferencia de las variables de entrada para el espacio de salida, escogemos cinco conjuntos borrosos. El número queda marcado por el número de reglas que se tienen en mente realizar. Las etiquetas de esta variable de control son im: izquierda mucho, i: izquierda, m: mantenerse, b: bajar, bm: bajar mucho. En la siguiente figura se muestra el rango de valores y sus particiones con las cinco funciones de pertenencia.



Conjuntos Borrosos de la variable de salida M para el yaw

Los valores de las ganancias obtenidas ante ensayo y error han sido:

$$GE = 1$$

$$GDE = 3$$

$$GIE = 5$$

$$GU = 1$$

En la tabla 4 se muestra las acciones de control para los tres posibles valores de entrada que hay. Aunque el número de reglas teóricas es 3^3 , se han podido agrupar algunas acciones en una sola regla ya que la acción de control es la misma reduciendo el número de reglas a once.

DeYaw - IeYaw \ eYaw	Shorario	Mantenerse	SAntihorario
Negativa-Negativa	dm	d	i
Negativa-Igual	dm	d	i
Negativa-Positiva	dm	d	i
Igual-Negativa	d	d	i
Igual-Igual	d	m	i
Igual-Positiva	d	i	i
Positiva-Negativa	d	i	im
Positiva-Igual	d	i	im
Positiva-Positiva	d	i	im

Tabla 4: Reglas de Inferencia de el yaw

Resultados

En esta sección veremos un conjunto reducido pero significativo de los resultados que se han ido obteniendo. Primeramente se verá la simulación de movimientos individuales. Luego se verán pruebas en la que se producen varios movimientos de manera simultanea.

4.1. Resultados Individuales

Se han reflejado dos pruebas por cada movimiento:

Para la altura reflejarán los resultados de bajar y subir.

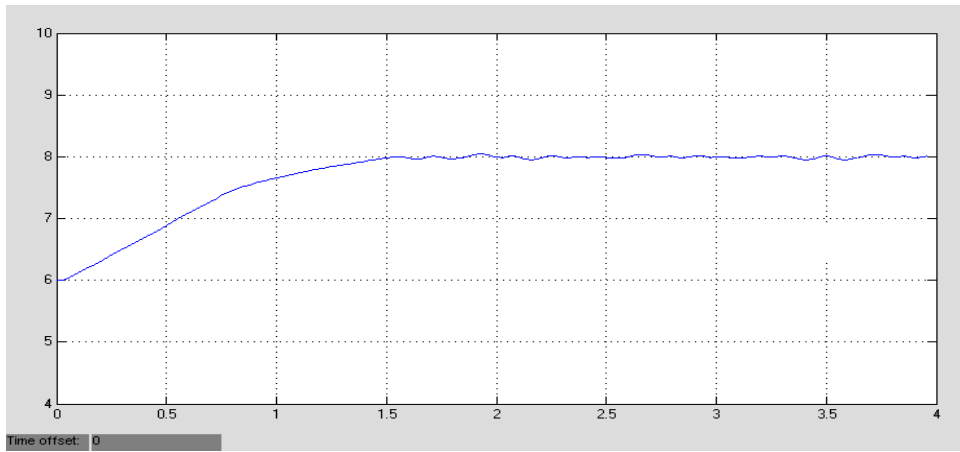
Para el roll los resultados de inclinarse a la izquierda y girar a la derecha.

Para el pitch los resultados de ir para delante y para atrás.

Para el yaw los resultados de girar a la izquierda y a la derecha.

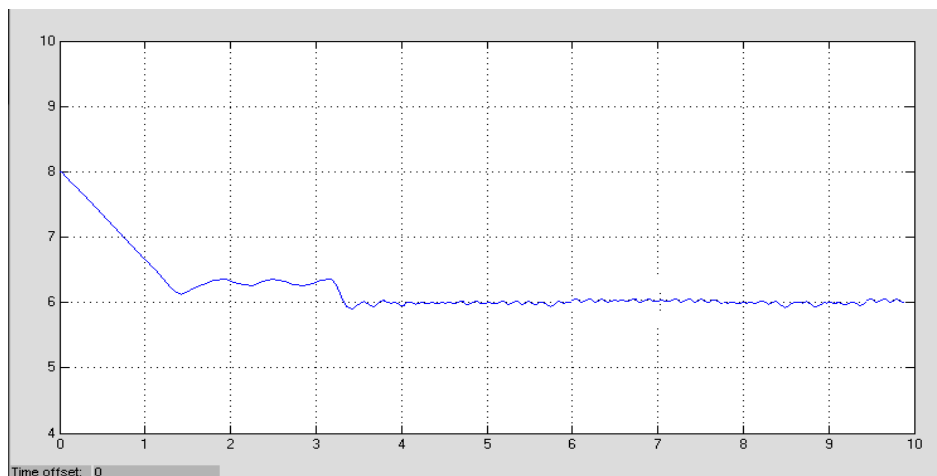
4.1.1. Resultados de la Altura

Los resultados para la altura han sido los siguientes:



Simulación de la altura al subir de 6 a 8 con controlador Fuzzy PID

En la figura 1 se observa como la subida es suave tardando dos segundos en alcanzar el valor deseado.



Simulación de la altura al bajar de 8 a 6 con controlador Fuzzy PID

El quadrotor es un vehículo que debería actuar con suavidad, especialmente en la bajada, en la que no es conveniente que nos pasáramos y rebotáramos contra el suelo al intentar aterrizar.

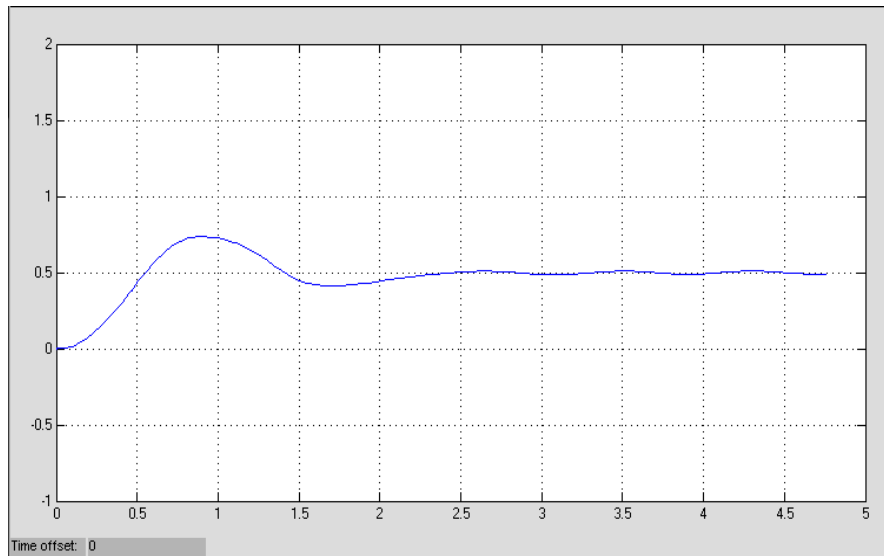
En la figura 2 se ve como en la bajada se queda un poco por encima del valor deseado hasta que a los cuatro segundos se ajusta. Error que se corrige gracias a la integral y que es producido por la baja aportación de la derivada. Aumentando su ganancia (GDE) se arregla este problema, pero la derivada introduciría ruidos que afectan negativamente al resto de variables empeorando la situación.

4.1.2. Resultados del Roll

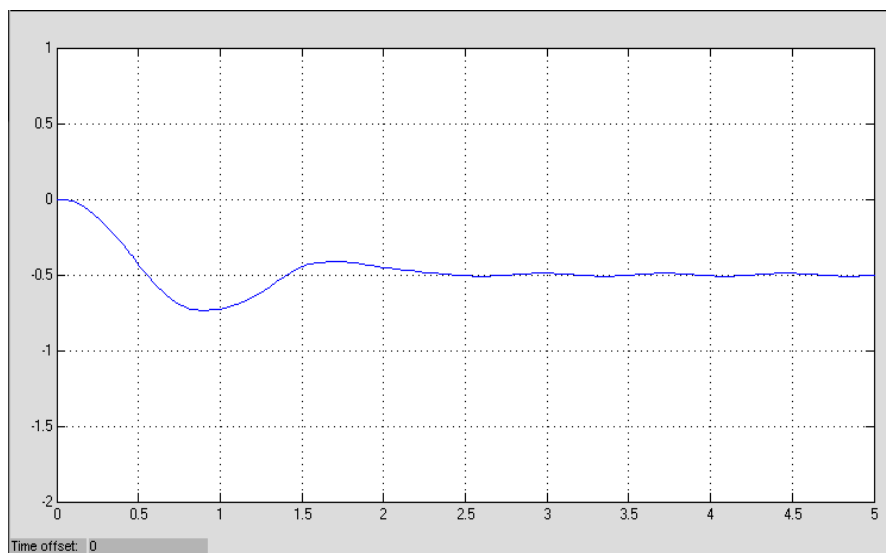
A continuación se ven unos resultados para giros cerca del límite (0.5 radian), en el que se aprecia que el tiempo de respuesta es pequeño. Aparecen unos pequeños overshoots al principio, pero que se le ha dado menos importancia que el que pudiera ocurrir en la altura; ya que aquí, que se gire un

poco más no es demasiado preocupante, a no ser que provoque una desestabilización en el quadrotor. difícil porque para eso se han puesto limitaciones en el giro, además que no serían corrientes giros tan bruscos.

Lo importante es que en el tiempo de estabilización no se produzcan balanceos violentos y como se puede observar esos balanceos no ocurren.



Simulación del Roll de 0 a 0.5 radian con controlador Fuzzy PID



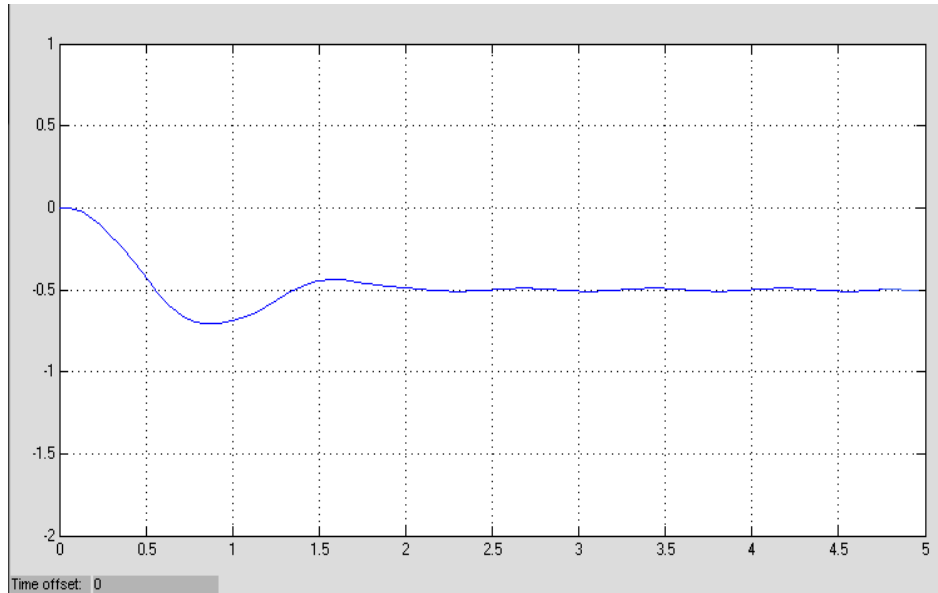
Simulación del Roll de 0 a -0.5 radian con controlador Fuzzy PID

4.1.3. Resultados del Pitch

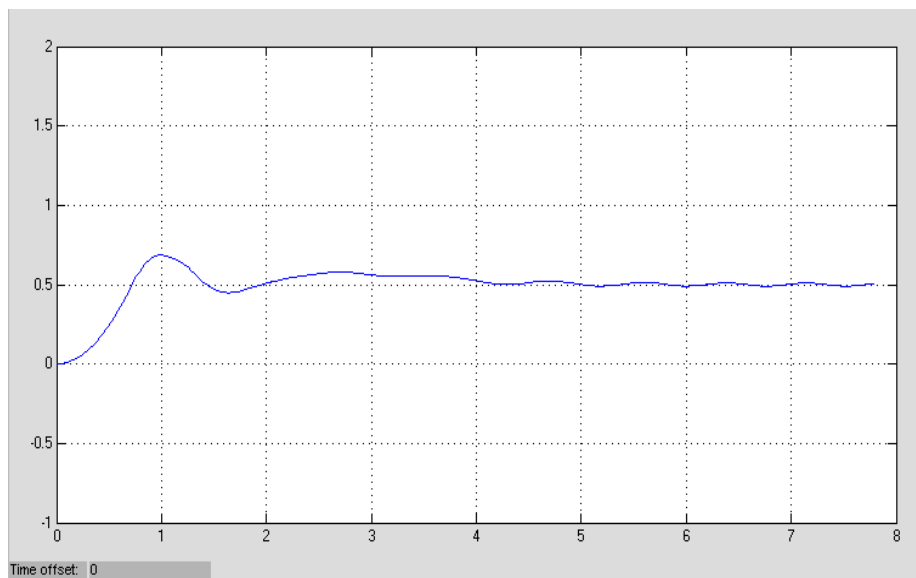
A continuación se ven unos resultados para giros de 0.5 radian, en el que se aprecia que el tiempo

de respuesta es pequeño. Aparecen unos pequeños oversums al principio, pero igualmente tienen menos importancia que el que pudiera ocurrir en la altura.

En el tiempo de estabilización no se producen balanceos violentos teniendo un error que no supera nunca 1° .



Simulación del pitch de 0 a -0.5 radian con controlador Fuzzy PID



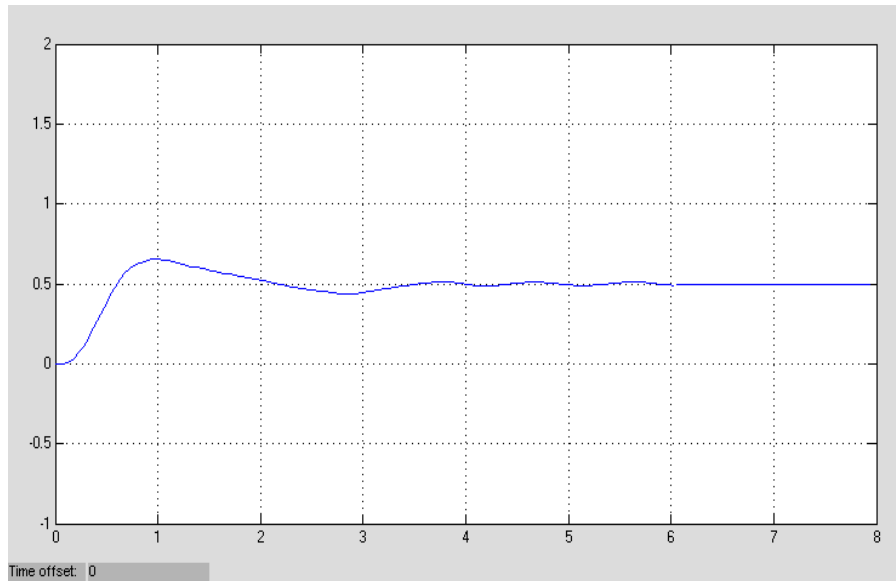
Simulación del pitch de 0 a 0.5 radian con controlador Fuzzy PID

4.1.4. Resultados del Yaw

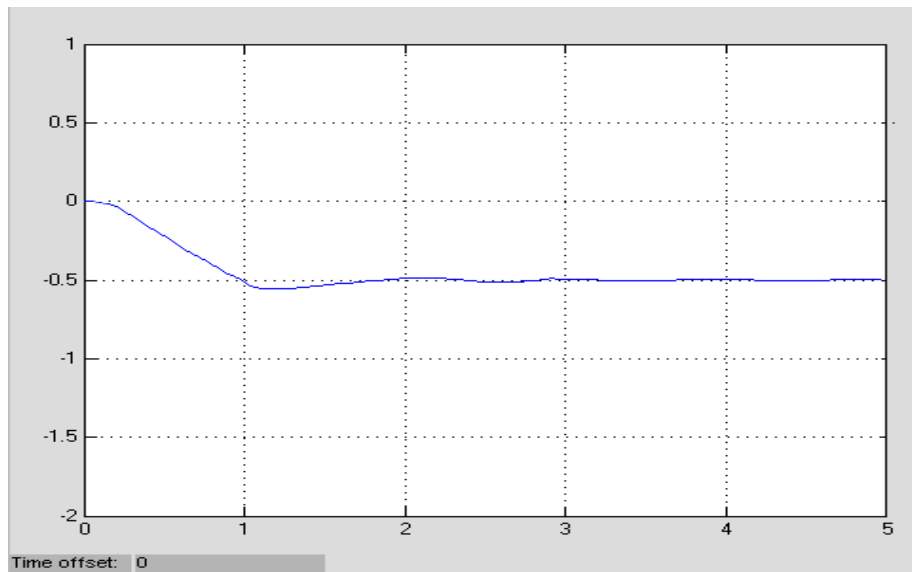
A continuación se ven unos resultados para giros de 0.5 radian, en el que se aprecia que el tiempo de respuesta es pequeño. Aparecen unos pequeños oversums al principio, que son corregidos de

manera muy suave sin mayor incidencias.

Al igual que en los otros dos giros en el tiempo de estabilización no se producen sobreoscilaciones.



Simulación del yaw de 0 a 0.5 radian con controlador PID



Simulación del yaw de 0 a -0.5 radian con controlador Fuzzy PID

4.2. Resultados Combinados

En esta sección se detallaran pruebas de vuelo que simulan el control de todas las variables que hemos tratado de manera simultanea (altura, roll, pitch, yaw).

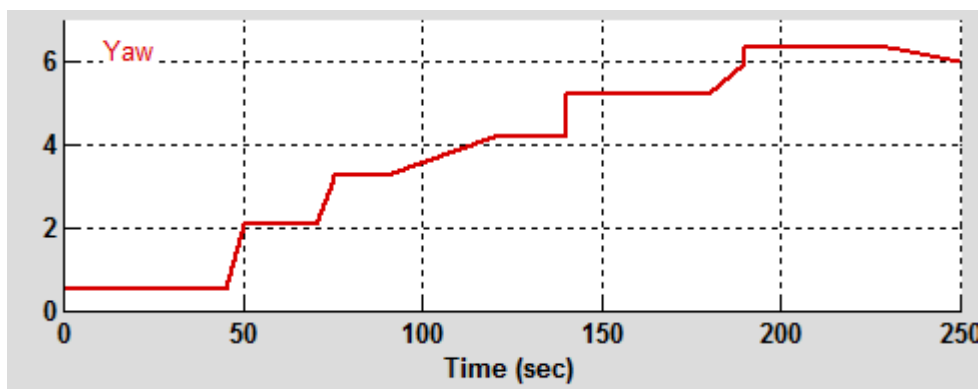
El simulador además incorpora un plano X-Y en el que se visualiza la ruta propuesta. En él se pueden observar los movimientos laterales ocasionados por el roll o el yaw y los movimientos de avance producidos por el pitch.

4.2.1. Primera Prueba

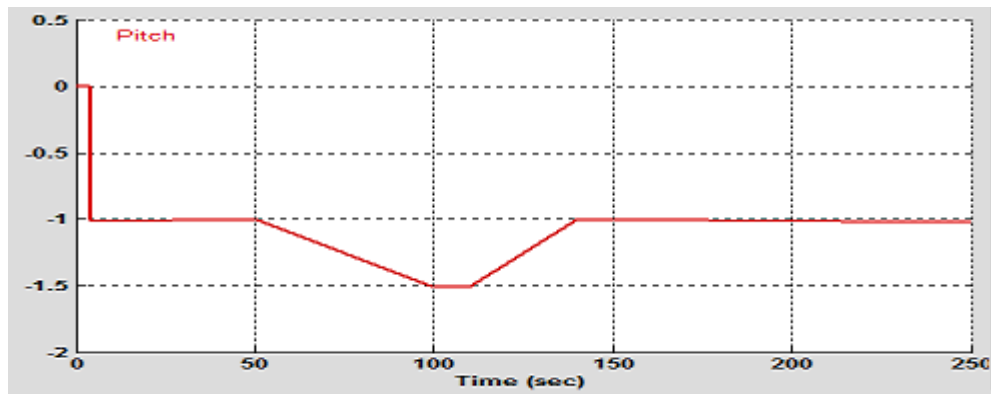
En la primera prueba se van a analizar la altura, el pitch y el yaw de manera conjunta. La altura sólo será variada al principio en el despegue, el pitch y el yaw cambiarán de valor con frecuencia con los dos generadores de señales que se ven en las siguientes figuras. Estos giros permitirán el avance del vehículo y giros con el yaw de mayor o menor ángulo. También se comprobará como actúa el sistema ante decisiones que pudieran provocar desestabilizaciones, como es el caso de una inclinación excesiva del quadrotor hacia delante.

En la situación inicial el quadrotor está en el suelo y con un ángulo de yaw de 0.5 radianes ($28,66^\circ$).

El quadrotor despegará y subirá hasta alcanzar los veinte metros conservando el ángulo de yaw inicial. A los cuatro segundos se produce un movimiento en el pitch de -1 radian (figura 10) que provocará que el vehículo avance. A los cuarenta y cinco segundos el quadrotor tiene que girar a la izquierda hasta alcanzar los 2,1 radianes ($120,435^\circ$) en los cincuenta y uno segundos (figura 9). Además a los cincuenta segundos se acelera suavemente hasta que llegue a -1.5 radianes a los cien segundos. Como es un giro excesivo, y el límite es -1.3, radianes el sistema no permitirá que alcance a -1.5 (figura 12). Con esa velocidad máxima el quadrotor seguirá avanzando hasta que llegue a los setenta segundos en el que el quadrotor empezará a girar a la izquierda hasta los 3,85 radianes que se alcanzarán cinco segundos después (figura 9). En esa situación se mantendrá hasta que en el segundo noventa se produzca un nuevo giro a la izquierda que llegué a una posición de 4,2 radianes a los ciento veinte segundos. Mientras en el ciento diez se produce una disminución progresiva de la velocidad al pasar de -1.3 radianes a -1 radian a los ciento cuarenta segundos (figura 10). A partir de ahí el pitch se mantendrá constante avanzando el vehículo a la misma velocidad y sólo se producirán cambios en el yaw que girará alguna vez más a la izquierda, antes de pararse la ejecución a los doscientos cincuenta segundos.



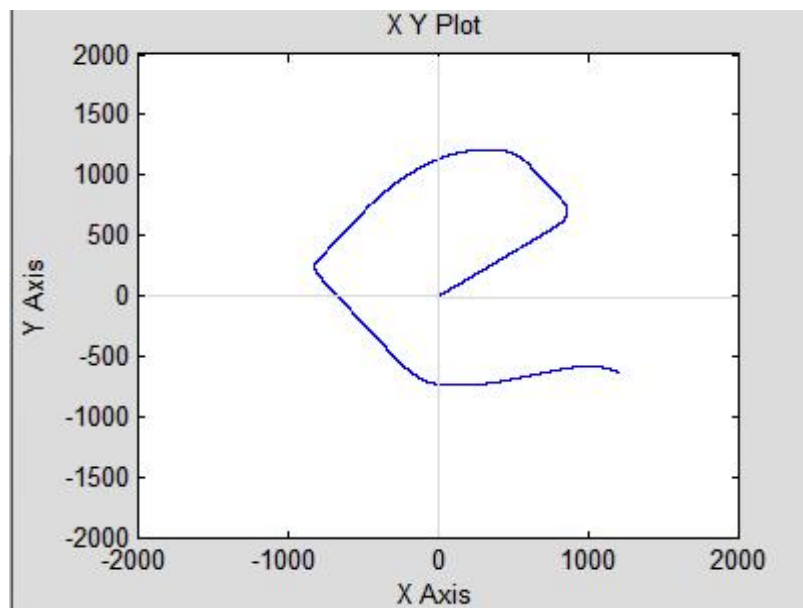
Generador de Señales de Yaw para la primera prueba



Generador de Señales de pitch para la primera prueba

Ahora comprobamos como la ruta marcada e indicada se cumple visualizándolo tanto en el plano X-Y como en el cuadro de todos los controles.

La ruta resultante en el plano X-Y es la de la figura 11. Comienza en el punto (0, 0) y la forma de e es consecuencia de los giros a la izquierda que se han comentado y visto en la figura 9. Los trazados suaves se corresponden con los giros en el que se da bastante tiempo para hacer el giro (pendientes pequeñas en figura 9). Los giros más bruscos se corresponden en las acciones que demandaban ser más rápidas (pendientes grandes en figura 9).



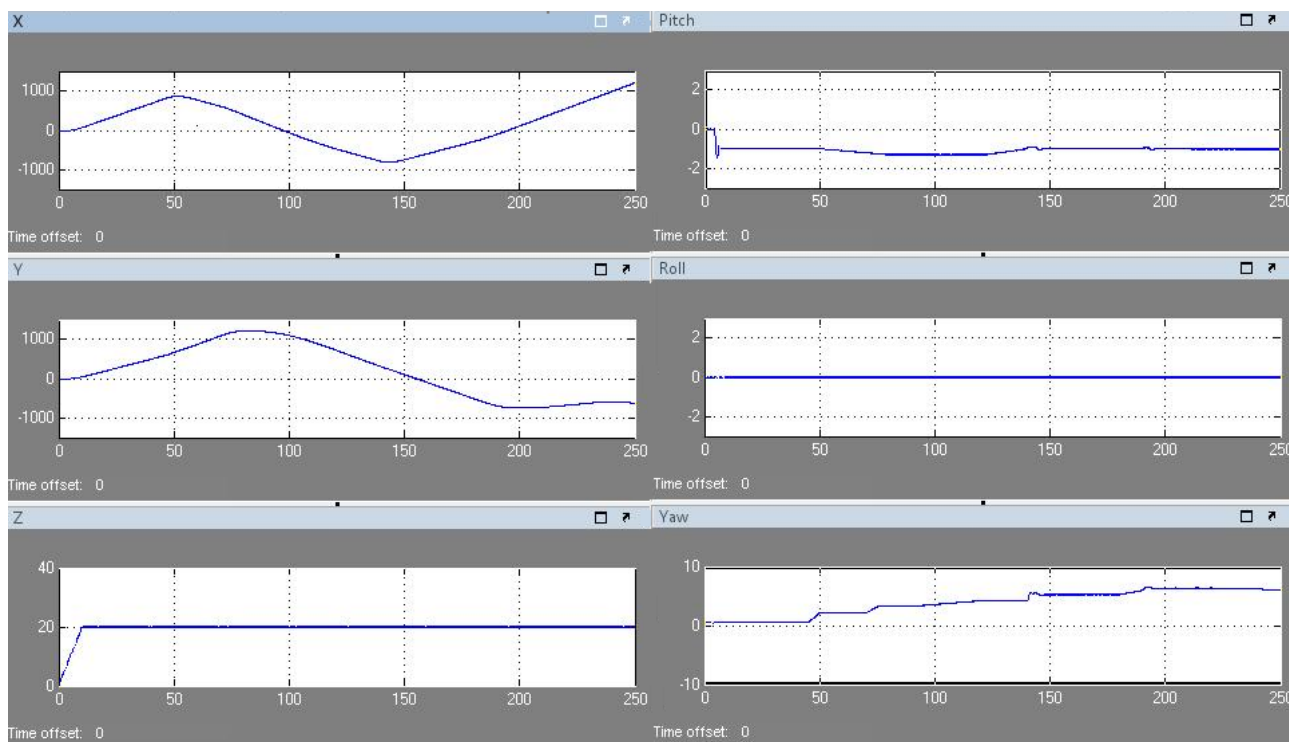
Plano X-Y de la primera prueba

Finalmente se adjunta un cuadro en el que se ven las acciones de control sobre las tres coordenadas espaciales (x,y,z) y sobre los tres giros (pitch, Roll y Yaw). Ver Figura 4.

En el caso de roll se ve como no se actúa sobre él y permanece estable en el cero. En el caso de la altura (Z) se comprueba como sube rápidamente hasta los veinte metros y ahí se mantiene hasta que finalice la ejecución.

En el caso de los movimientos pitch y yaw se puede comparar como se ejecutan los movimientos deseados en la ruta marcada en las figuras 1 y 2 a tiempo. Además en el caso del pitch se puede ver como no permite que llegue a los -1.5 radianes, funcionando perfectamente el límite inferior en -1.3 radianes.

Finalmente en el caso de la X y la Y se puede ver las acciones vistas en el plano X-Y pero con la referencia de tiempo. Además se puede ver como los cambios en el Yaw provocan cambios en estas dos gráficas.



Cuadro visualizador de control de la primera prueba

4.2.2. Segunda Prueba

En la segunda prueba se van a analizar la altura, el pitch, el roll y el yaw de manera conjunta.

En lo relativo a la altura esta vez habrá despegue, vuelo estacionario y aterrizaje. El pitch, roll y yaw cambiarán de valor con frecuencia con los tres generadores de señales que se ven en las siguientes figuras. Estos giros permitirán el avance, frenado y marcha atrás del vehículo, además se producirán desplazamientos laterales con el roll y giros con el yaw. Se verá también como actúan las limitaciones en el roll.

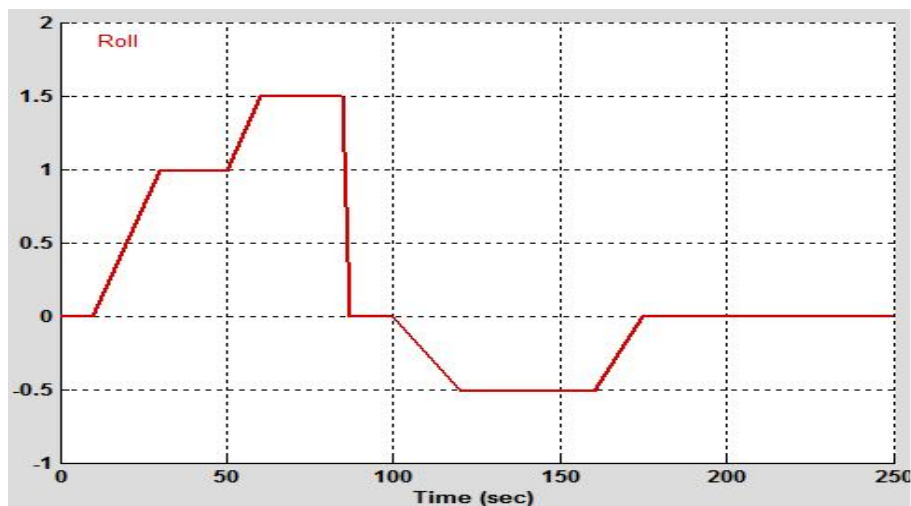
En la situación inicial el quadrotor está en el suelo y con todos los ángulos de giro a cero.

El quadrotor despegará y subirá hasta alcanzar los veinte metros de altura sin ningún giro. A los cuatro segundos se producirá un movimiento en el pitch de -0.8 radianes (figura 12) que provocará que el vehículo avance. A los diez segundos empieza a actuar el roll, inclinándose hacia la izquierda al principio de manera suave y luego con una mayor inclinación hasta que alcance el límite de inclinación permitido, 1.3 radianes (74,5°) a los aproximadamente cincuenta y siete segundos, lo

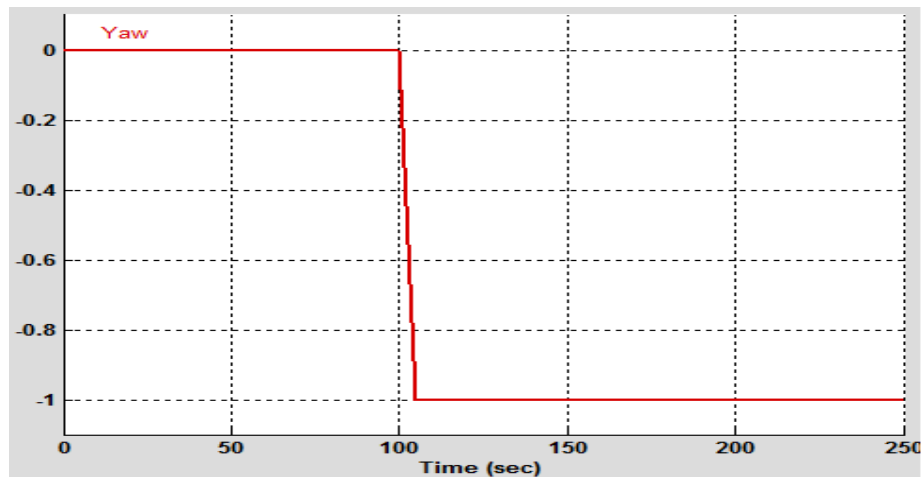
que impedirá que se llegué al valor fuera del límite deseado, 1.5 radianes, (figura 13). El quadrotor seguirá avanzando inclinado hasta que en el segundo ochenta y siete el quadrotor se reincorporará y sólo actuará el pitch durante un período de tránsito que durará hasta los cien segundos. En ese momento empezará a actuar simultáneamente el roll (-0.5 radianes) y el yaw (-1 radian) ambos con giros a la derecha. el yaw (figura 14) permanecerá con ese ángulo hasta el final de la ejecución conservando el rumbo de (-1 radian), aunque influido por el roll que seguirá inclinado -0.5 radianes a la derecha hasta los ciento sesenta segundos. En ese instante el quadrotor seguirá moviéndose hacia delante pero ya irá frenando y a los doscientos segundos el quadrotor dará marcha atrás hasta los doscientos veinte segundos. En el que no se producirá ningún giro para que a los doscientos treinta segundos inicie el aterrizaje, llegue al suelo y finalice la ejecución.



Generador de Señales de pitch para la segunda prueba



Generador de Señales de Roll para la segunda prueba



Generador de Señales de Yaw para la segunda prueba

La ruta resultante en el plano X-Y de esta segunda prueba es la de la figura 8. Comienza en el punto (0, 0) y puede dividirse en cinco trazados significativos.

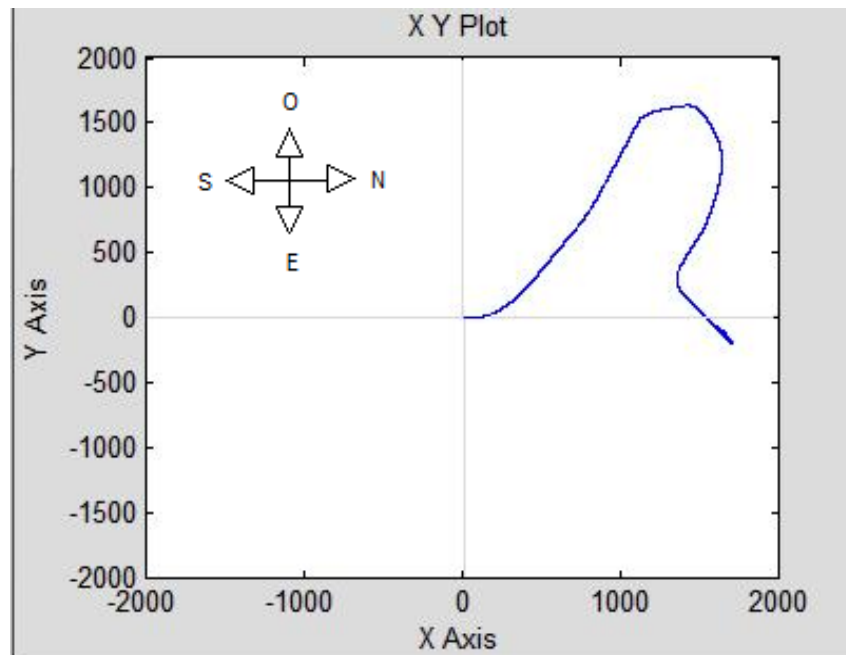
El primer trazado es el que se corresponde con la actuación de manera simultanea del pitch y roll. El quadrotor ha despegado, alcanzado los veinte metros de altura y avanzado unos metros hacia delante. Pero al empezar la influencia del pitch se observa como produce la primera curva hacia la izquierda. Hay que recordar que el roll por si sólo no produce un giro en la cabeza del vehículo, sino simplemente un desplazamiento lateral, pero siempre con la misma orientación del vehículo con el que se empezó. Al principio debido a que el roll es pequeño se avanza más por el eje de la X que el de la Y, pero al final de la curva se ve que se ha avanzado más de 1500 metros en el eje de las Y por apenas 1000 metros en el eje de las X debido a que el roll es entonces más pronunciado que el pitch.

En el segundo trazado se observa un cambio de dirección en el que se avanza en el eje de las X y no en el de las Y debido a que el roll deja de actuar y el quadrotor ya sólo está inclinado hacia delante para avanzar hacia delante.

En el tercer trazado se ve una curva cerrada producida por el giro de el yaw de 57° (1 radian) hacia la derecha, que provoca que el quadrotor se desplace hacia el noreste. El roll a la derecha tarda un poco más en actuar debido a que su giro es más suave (menor pendiente figura 16), pero cuando hace efecto sí que consigue que el trazado se desplace a la derecha modificando el rumbo.

El cuarto trazado en el que aparece un giro a la izquierda no es más que la consecuencia de que el roll ha dejado de actuar y ya no se produce ese desplazamiento lateral sino que sólo se avanza hacia delante con la orientación de 57° Noreste marcado por el yaw.

Finalmente en el quinto trazado es el correspondiente a la última parte en el que se ve como tras avanzar un rato en la dirección comentada anteriormente el trazado es más grueso al final porque el vehículo va marcha atrás antes de aterrizar y pararse la ejecución definitivamente.



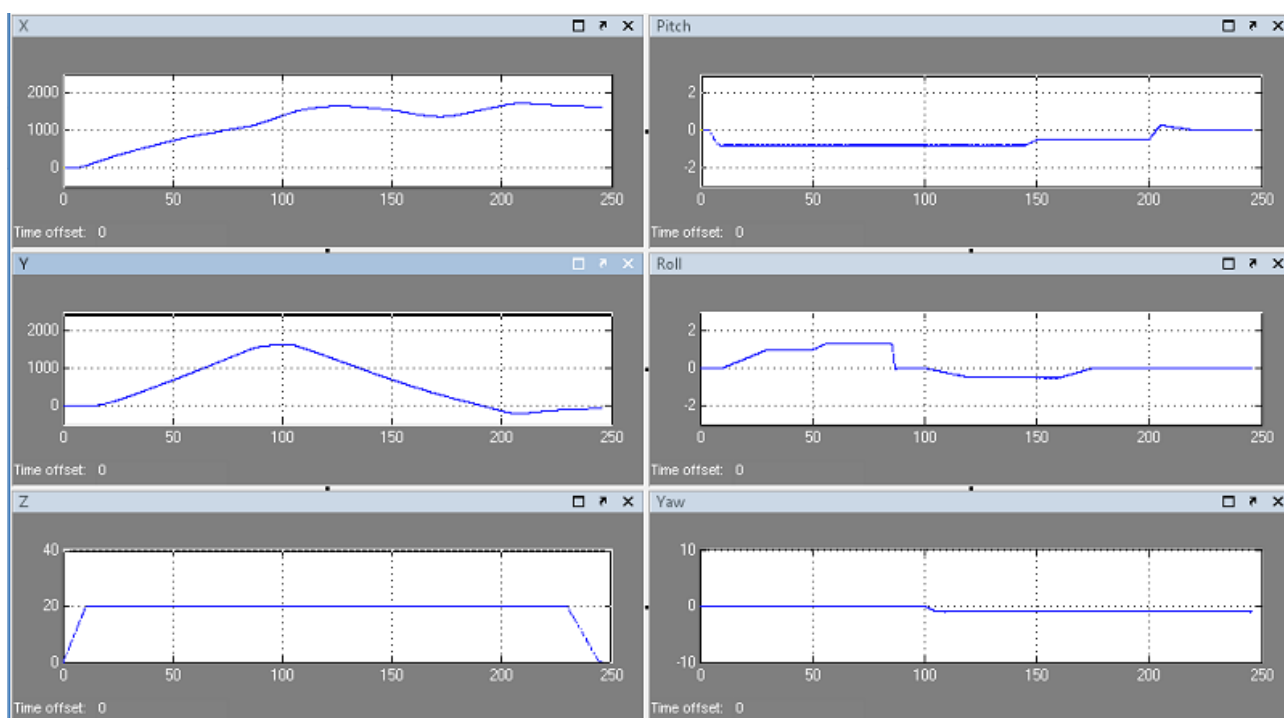
Plano X-Y de la segunda prueba

Finalmente se adjunta el cuadro de control en el que se ven las acciones simultáneamente sobre las tres coordenadas espaciales (x,y,z) y sobre los tres giros (pitch, Roll y Yaw) que responde el sistema. Ver Figura 17.

En el caso de la altura (Z) se comprueba como sube rápidamente de cero a veinte metros y ahí se mantiene hasta que al final de la ejecución aterrizan.

En el caso de los movimientos Pitch, Roll y Yaw se puede comparar como se ejecutan los movimientos deseados en la ruta marcada en las figuras 5, 6 y 7 respectivamente. Además en el caso del roll se puede ver como no permite que llegue a los -1.5 radianes, funcionando perfectamente el límite inferior en -1.3 radianes.

Finalmente en el caso de la X y la Y se puede ver las acciones vistas en el plano X-Y pero con la referencia de tiempo. Conviene resaltar el efecto del Yaw en la señal Y cuando empieza a actuar en los cien segundos. También se ve bien como cuando el pitch al final es positivo se empieza a ir para atrás provocando esos cambios en las gráficas X e Y.



Cuadro visualizador de control de la segunda prueba

Conclusiones y trabajos futuros

La aplicación de la lógica difusa en el diseño de los sistemas de control, se presenta como un método de gran capacidad y eficacia. Aprovechando el conocimiento del experto que conoce el comportamiento del sistema a controlar, permite realizar toma de decisiones similares a como la haría un humano pero con aproximaciones numéricas.

Los resultados obtenidos en este trabajo han sido satisfactorios, puesto que se ha conseguido controlar de manera robusta y estable las señales de controlar, satisfaciendo el clásico compromiso de diseño estabilidad-precisión-velocidad de respuesta.

No obstante, utilizando adicionalmente algún algoritmo de autosintonía, podría mejorar el sistema reduciendo aún más las oscilaciones y tiempos de establecimiento provocados por la dependencia entre las variables ajustadas. Añadiendo un módulo nuevo supervisor, capaz de corregir las perturbaciones estimando los nuevos parámetros que reajusten al controlador de manera inmediata.

Sería muy interesante ver el comportamiento del modelo utilizando una maqueta que está disponible en la Universidad para tal efecto. Se sabe que esto conllevará problemas de ruidos y en las señales de retroalimentación producidas de los sensores.

Bibliografía

- [1] Bouabdallah, A. Noth and R. Siegwart. *Dynamic Modelling of UAV*, 2006.École Polytechnique fédérale de Lausanne.
- [2] Castillo,P. , P.Garcia, R.Lozano, P.Albertos, 2007. *Modelado y estabilización de un helicóptero con cuatro rotores*. RIAI, vol. 4, n.1, pp. 41-57.
- [3] Diaz J.L., A.Pardo, A.Muñoz, 2007. *Control fuzzy de inducción controlado por campo orientado*, Tecnología Avanzada, vol. 1, n. 9, ISSN 1692-7257.
- [4] Dormido, S., M.Santos, A.Pérez, F.Morrilla. *Autosintonía de controladores borrosos utilizando técnicas clásicas basadas en reguladores PID*. Proc. of III FLAT, España, 1993, pp217-225.
- [5] Li-Xing Wang. *A course in Fuzzy Systems and Control*, 1997.
- [6] Mandami, E. M., Applications of fuzzy algorithms for control of simple dynamics plant. Proc. IEEE 121 (1974) 1585-1588
- [7] Ñeco R.P., Reinoso O., Garcia N., Aracil R. *Apuntes de Sistemas de control*, ECU 2003.
- [8] Ollero, A. and I.Maza. *Multiple Heterogeneous Unmanned Aerial Vehicles*. Vol.37/2007. Springer Berlin/Heidelberg.
- [9] Santos M. *Contribución a las técnicas de sintonía de los controladores borrosos en la lógica borrosa*. Ph. D. Dissertation, Departament of Computer Science. Universidad Complutense de Madrid, 1994.
- [10] Sanchez D. *Cinemática y Dinámica del Cuadro rotor*. UCM, 2008.
- [11] Sugeno et al. *Intelligent control of an unmanned helicopter based on fuzzy logic*, Proc of American Helicopter Society 51th Annual Forum, Texas, 1995.
- [12] Sugeno, M., *Industrial applications of fuzzy control*. Ed. Elsevier. North Holland, Amsterdam. 1985.
- [13] Stevens, Brian L. And F.L. Lewis. *Aircraft control and Simulation*. John Wiley & Sons, Inc.,2003, 2nd edition.